



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

Important Instructions to examiners:

- 1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
- 2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
- 3) The language errors such as grammatical, spelling errors should not be given more importance (Not applicable for subject English and Communication Skills).
- 4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
- 5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
- 6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
- 7) For programming language papers, credit may be given to any other program based one equivalent concept.

Q.N o.	Sub Q.N.	Answer	Marking Scheme
1.	(A) (a) Ans.	Attempt any THREE of the following: State and explain any four features of Java. <i>(Note: Any four may be considered)</i> i. Java is an object oriented language:- It follows all the principles of object oriented programming namely inheritance, polymorphism and abstraction. Multiple inheritance is possible with the concept of interface ii. Java is both compiled and interpreted:- Most of the programming languages either uses a compiler or an interpreter. Java programs are to be compiled to get an intermediate byte code (a .class file) and then interpreted making it more secure and platform independent. iii. Java is secure: <ul style="list-style-type: none">• Java does not use pointer.• Java programs run inside a virtual machine• Classloader adds security by separating the package for the classes of the local file system from those that are imported from network sources• Bytecode Verifier checks the code fragments for illegal code	3 x 4= 12 4M 1M for each feature



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<p>that can violate access right to objects</p> <ul style="list-style-type: none">• Security Manager determines what resources a class can access such as reading and writing to the local disk <p>iv. Robust: Java uses strong memory management. The lack of pointers avoids security problem. There is automatic garbage collection in java. There is exception handling and type checking mechanism in java</p> <p>v. Architecture-neutral: There is no implementation dependent features e.g. size of primitive types is fixed</p> <p>vi. Platform independent and Portable: java byte code can be carried to any platform</p> <p>vii. Distributed: Distributed applications can be created in java. RMI and EJB are used for creating distributed applications. We may access files by calling the methods from any machine on the internet</p> <p>viii. Multithreaded: A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications etc.</p>	
(b) Ans.	<p>Write any four methods of file class with their use. (Note: Any four methods may be considered)</p> <p>public String getName() Returns the name of the file or directory denoted by this abstract pathname.</p> <p>public String getParent() Returns the pathname string of this abstract pathname's parent, or null if this pathname does not name a parent directory.</p> <p>public String getPath() Converts this abstract pathname into a pathname string.</p> <p>public boolean isAbsolute() Tests whether this abstract pathname is absolute.</p> <p>public String getAbsolutePath() Returns the absolute pathname string of this abstract pathname.</p>	<p>4M</p> <p><i>1M each for method and use</i></p>	



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<p>public boolean canRead() Tests whether the application can read the file denoted by this abstract pathname.</p> <p>public boolean canWrite() Tests whether the application can modify the file denoted by this abstract pathname.</p> <p>public boolean exists() Tests whether the file or directory denoted by this abstract pathname exists.</p> <p>public boolean isDirectory() Tests whether the file denoted by this abstract pathname is a directory.</p> <p>public boolean isFile() Tests whether the file denoted by this abstract pathname is a normal file.</p> <p>public boolean isHidden() Tests whether the file named by this abstract pathname is a hidden file</p> <p>public long lastModified() Returns the time that the file denoted by this abstract pathname was last modified.</p> <p>public long length() Returns the length of the file denoted by this abstract pathname</p> <p>public boolean createNewFile() throws IOException Atomically creates a new, empty file named by this abstract pathname if and only if a file with this name does not yet exist</p> <p>public boolean delete() Deletes the file or directory denoted by this abstract pathname</p> <p>public String[] list() Returns an array of strings naming the files and directories in the directory denoted by this abstract pathname.</p> <p>public boolean mkdir() Creates the directory named by this abstract pathname.</p>	
--	---	--



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>public boolean renameTo(File dest) Renames the file denoted by this abstract pathname. public boolean setLastModified(long time) Sets the last-modified time of the file or directory named by this abstract pathname. public boolean setReadOnly() Marks the file or directory named by this abstract pathname so that only read operations are allowed public boolean setWritable(boolean writable, boolean ownerOnly) Sets the owner's or everybody's write permission for this abstract pathname. public boolean equals(Object obj) Tests this abstract pathname for equality with the given object public String toString() Returns the pathname string of this abstract pathname.</pre>	
(c) Ans.	<p>Explain any two relational operators in Java with example.</p> <p>The relational operators in java are:</p> <p>< - This operator is used to check the inequality of two expressions. It returns true if the first expression is less than the second expression else returns false.</p> <pre>if(Exp1< exp2) { do this } else { do this }</pre> <p>>-This operator is also used to check the inequality of two expressions. It returns true if the first expression is greater than the second one else returns false.</p> <pre>if(Exp1> exp2) { do this } else { do this }</pre> <p><=- This operator returns true if the first expression is less than or equal to the second expression else returns false.</p>	<p>4M</p> <p><i>1M each for listing and explainin g any two relational operators</i></p> <p><i>1M each for program</i></p>	



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<pre>if(Exp1<=exp2) { do this } else { do this } >=-This operator returns true if the first expression is greater than or equal to the second expression else returns false. if(Exp1>= exp2) { do this } else { do this } == -This operator returns true if the values of both the expressions are equal else returns false. if(Exp1== exp2) { do this } else { do this } != - This operator returns true if the values of both the expressions are not equal else returns false. if(Exp1!= exp2) { do this } else { do this } Example: class RelationalOps { public static void main(String args[]) { int a = 10; int b = 20; System.out.println("a == b = " + (a == b)); System.out.println("a != b = " + (a != b)); System.out.println("a > b = " + (a > b)); System.out.println("a < b = " + (a < b)); System.out.println("b >= a = " + (b >= a)); System.out.println("b <= a = " + (b <= a)); } }</pre>	<p><i>(2M for one program with both the operators)</i></p>
--	---	--



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<p>(d)</p> <p>Ans.</p>	<p>What is thread? Draw thread life cycle diagram in Java. <i>(Note: Explanation of the life cycle is not needed).</i></p> <p>A thread is a single sequential flow of control within a program. They are lightweight processes that exist within a process. They share the process's resource, including memory and are more efficient. JVM allows an application to have multiple threads of execution running concurrently. Thread has a priority. Threads with higher priority are executed in preference to threads with lower priority.</p> <div style="text-align: center;"> <pre> graph TD NB((New Born)) -- start() --> ActiveThreadBox subgraph ActiveThreadBox [Active thread] R((Running)) RN((Runnable)) R -- scheduled() --> RN RN -- yield() --> R end ActiveThreadBox -- stop() --> D[Dead] ActiveThreadBox -- suspend() --> B[Blocked] ActiveThreadBox -- sleep() --> B ActiveThreadBox -- wait() --> B B -- resume() --> RN B -- notify() --> RN B -- stop() --> D </pre> </div>	<p>4M</p> <p><i>2M for defining a thread</i></p> <p><i>2M for diagram</i></p>
1.	<p>(B)</p> <p>(a)</p> <p>Ans.</p>	<p>Attempt any ONE of the following:</p> <p>What is single level inheritance? Explain with suitable example. <i>(Note: Any appropriate program may be written).</i></p> <p>Single level inheritance enables a derived class to inherit properties and behaviour from a single parent class. It allows a derived class to inherit the properties and behaviour of a base class, thus enabling code reusability as well as adding new features to the existing code. This makes the code much more elegant and less repetitive. Single level inheritance can be represented by the following</p>	<p>1 x 6 = 6 6M</p> <p><i>Explanation with suitable diagram</i> 2M</p>

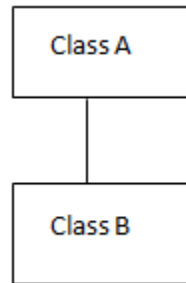


MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515



Example:

```
class SingleLevelInheritanceParent {
    int l;
    SingleLevelInheritanceParent(int l) {
        this.l = l;
    }
    void area() {
        int a = l*l;
        System.out.println("Area of square :"+a);
    }
}

class SingleLevelInheritanceChild extends SingleLevelInheritanceParent
{
    SingleLevelInheritanceChild(int l) {
        super(l);
    }
    void volume() {
        int v;
        v= l*l*l;
        System.out.println("Volume of the cube is "+v);
    }
    void area() {
        int a;
        a = 6*l*l;
        System.out.println("Total surface area of a cube is "+a);
    }
}

class SingleLevelInheritance {
```

**4M for
correct
program**



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>public static void main(String ar[]) { SingleLevelInheritanceChild cube = new SingleLevelInheritanceChild(2); cube.volume(); cube.area(); }</pre>	
(b)	What is package? State how to create and access user defined package in Java. <i>(Note: Code snippet can be used for describing)</i>		6M
Ans.	<p>Package is a name space that organizes a set of related classes and interfaces. Conceptually, it is similar to the different folders in a computer. It also provides access protection and removes name collision.</p> <p>Packages can be categorized into two:- built-in and user defined.</p> <p>Creation of user defined package: To create a package a physical folder by the name should be created in the computer. Example: we have to create a package myPack, so we create a folder d:\myPack The java program is to be written and saved in the folder myPack. To add a program to the package, the first line in the java program should be package <name>; followed by imports and the program logic.</p> <pre>package myPack; import java.util; public class Myclass { //code }</pre> <p>Access user defined package: To access a user defined package, we need to import the package in our program. Once we have done the import we can create the object of the class from the package and thus through the object we can access the instance methods.</p> <pre>import mypack.*; public class MyClassExample{ public static void main(String a[]) { Myclass c= new Myclass(); } }</pre>	<p>2M for definition of package</p> <p>2M each for explanation of creation and access of user defined package</p>	



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

2.	(a) Ans.	Attempt any TWO of the following: Write a program to add 2 integer, 2 string and 2 float objects to a vector. Remove element specified by user and display the list. <pre>import java.util.*; import java.io.*; class Vect { public static void main(String a[]) { Vector<Object> v = new Vector<Object>(); v.addElement(new Integer(5)); v.addElement(new Integer(10)); v.addElement(new String("String 1")); v.addElement(new String("String 2")); v.addElement(new Float(5.0)); v.addElement(new Float(6.7)); int n=0; BufferedReader b = new BufferedReader(new InputStreamReader(System.in)); System.out.println("Following are the elements in the vector"); for(int i = 0; i <v.size();i++) { System.out.println("Element at "+i+ "is "+v.elementAt(i)); } System.out.println("Enter the position of the element to be removed"); try { n = Integer.parseInt(b.readLine()); } catch(Exception e) { System.out.println("Exception caught!" +e); } System.out.println("The element at "+n +"is "+v.elementAt(n)+" will be removed"); v.removeElementAt(n); System.out.println("The following are the elements in the vector"); for(int i = 0; i<v.size();i++) { System.out.println(v.elementAt(i)); } } }</pre>	2 x 8=16 8M <i>4M for correct syntax</i> <i>4M for correct logic</i>
	(b)	What is meant by interface? State its need and write syntax and features of interface.	8M



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	Ans.	<p>Interface is the mechanism by which multiple inheritance is possible in java. It is a reference type in Java. An interface has all the methods undefined. For a java class to inherit the properties of an interface, the interface should be implemented by the child class using the keyword “implements”. All the methods of the interface should be defined in the child class.</p> <p><i>Example:</i></p> <pre>interface MyInterface{ int strength=60; void method1(); void method2(); } public class MyClass implements MyInterface { int total; MyClass(int t) { total = t; } public void method1() { int avg = total/strength; System.out.println("Avg is "+avg); } public void method2() { } public static void main(String a[]) { MyClass c = new MyClass(3600); c.method1(); } }</pre> <p>Need: A java class can only have one super class. Therefore for achieving multiple inheritance, that is in order for a java class to get the properties of two parents, interface is used. Interface defines a set of common behaviours. The classes implement the interface, agree to these behaviours and provide their own implementation to the behaviours.</p>	<p><i>Definition</i> 2M</p> <p><i>program</i> <i>optional</i></p> <p>Need 2M</p>
--	------	--	--



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<p><i>Syntax:</i></p> <pre>interface InterfaceName { int var1 = value; int var2 = value; public return_type methodname1(parameter_list) ; public return_type methodname2(parameter_list) ; }</pre> <p>Features: Interface is defined using the keyword “interface”. Interface is implicitly abstract. All the variables in the interface are by default final and static. All the methods of the interface are implicitly public and are undefined (or implicitly abstract). It is compulsory for the subclass to define all the methods of an interface. If all the methods are not defined then the subclass should be declared as an abstract class.</p>	<p><i>Syntax</i> 2M</p> <p><i>Features</i> 2M</p>
<p>(c) Ans.</p>	<p>Explain applet life cycle with suitable diagram.</p> <pre>graph TD Start(()) -- init() --> Born((Born)) Born -- start() --> Running((Running)) Running -- paint() --> Running Running -- stop() --> Idle((Idle)) Idle -- start() --> Running Idle -- destroy() --> Dead((Dead))</pre> <p>Applets are small applications that are accessed on an Internet server, transported over the Internet, automatically installed, and run as part of a web document. The applet states include:</p> <ul style="list-style-type: none">• Born or initialization state• Running state• Idle state	<p>8M</p> <p><i>3M for diagram</i></p> <p><i>5M for explanation</i></p>



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<ul style="list-style-type: none">• Dead or destroyed state <p>Initialization state: applet enters the initialization state when it is first loaded. This is done by calling the init() method of Applet class. At this stage the following can be done:</p> <ul style="list-style-type: none">• Create objects needed by the applet• Set up initial values• Load images or fonts• Set up colours <p>Initialization happens only once in the life time of an applet.</p> <pre>public void init() { //implementation }</pre> <p>Running state: applet enters the running state when the system calls the start() method of Applet class. This occurs automatically after the applet is initialized. start() can also be called if the applet is already in idle state. start() may be called more than once. start() method may be overridden to create a thread to control the applet.</p> <pre>public void start() { //implementation }</pre> <p>Idle or stopped state: an applet becomes idle when it is stopped from running. Stopping occurs automatically when the user leaves the page containing the currently running applet. stop() method may be overridden to terminate the thread used to run the applet.</p> <pre>public void stop() { //implementation }</pre> <p>Dead state: an applet is dead when it is removed from memory. This occurs automatically by invoking the destroy method when we quit the browser. Destroying stage occurs only once in the lifetime of an applet. destroy() method may be overridden to clean up resources like threads.</p> <pre>public void destroy() { //implementation }</pre> <p>Display state: applet is in the display state when it has to perform some output operations on the screen. This happens after the applet</p>	
--	---	--



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		enters the running state. paint() method is called for this. If anything is to be displayed the paint() method is to be overridden. <pre>public void paint(Graphics g) { //implementation }</pre>	
3.	(a)	Attempt any FOUR of the following: Explain the following methods of string class with syntax and example: (i) substring() (ii) replace() <i>(Note: Any other example can be considered)</i> Ans. (i) substring(): Syntax: String substring(intstartindex) startindex specifies the index at which the substring will begin. It will return a copy of the substring that begins at startindex and runs to the end of the invoking string (OR) String substring(intstartindex,intendindex) Here startindex specifies the beginning index, and endindex specifies the stopping point. The string returned contains all the characters from the beginning index, upto, but not including, the ending index. Example : <pre>System.out.println(("Welcome".substring(3)); //come System.out.println(("Welcome".substring(3,5)); //co</pre> (ii) replace(): This method returns a new string resulting from replacing all occurrences of oldChar in this string with newChar. Syntax: String replace(char oldChar, char newChar) Example: <pre>String Str = new String("Welcome"); System.out.println(Str.replace('o', 'T')); // WelcTme</pre>	4 x 4 =16 4M <i>Each method syntax 1M and example 1M</i>
	(b)	Write a program to find sum of digit of number entered by user. <i>(Note: Direct Input or User Defined Input is also allowed & Any Other Logic also allowed)</i> Ans. class Sum1	4M



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<pre>{ public static void main(String args[]){ intnum = Integer.parseInt(args[0]); //takes argument as command line int remainder, result=0; while(num>0) { remainder = num%10; result = result + remainder; num = num/10; } System.out.println("sum of digit of number is : "+result); } }</pre> <p style="text-align: center;">OR</p> <pre>import java.io.*; class Sum11{ public static void main(String args[])throws IOException{ BufferedReaderobj = new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter number: "); int num=Integer.parseInt(obj.readLine()); int remainder, result=0; while(num>0) { remainder = num%10; result = result + remainder; num = num/10; } System.out.println("sum of digit of number is : "+result); } }</pre>	<p><i>Logic</i> 2M</p> <p><i>Syntax</i> 2M</p>
(c) Ans.	<p>What is Iterator class? Give syntax and use of any two methods of Iterator class.</p> <p>Iterator enables you to cycle through a collection, obtaining or removing elements.</p> <p>Each of the collection classes provides an iterator() method that returns an iterator to the start of the collection. By using this iterator object, you can access each element in the collection, one element</p>	<p>4M</p> <p><i>Definition</i> 1M</p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		at a time Syntax : Iterator iterator_variable = collection_object.iterator(); Methods: 1. Boolean hasNext(): Returns true if there are more elements. Otherwise, returns false. 2. Object next(): Returns the next element. Throws NoSuchElementException if there is not a next element. 3. void remove(): Removes the current element. Throws IllegalStateException if an attempt is made to call remove() that is not preceded by a call to next().	<i>Syntax 1M</i> <i>Any 2 methods 1M each</i>
	(d) Ans.	Describe the following attributes of applet. (i) Codebase (ii) Alt (iii) Width (iv) Code (i) Codebase: Codebase is an optional attribute that specifies the base URL of the applet code or the directory that will be searched for the applet's executable class file. (ii) Alt: Alternate Text. The ALT tag is an optional attribute used to specify a short text message that should be displayed if the browser cannot run java applets. (iii) Width: Width is a required attribute that gives the width (in pixels) of the applet display area. (iv) Code: Code is a required attribute that gives the name of the file containing your applet's compiled class file which will be run by web browser or appletviewer	4M <i>Each attribute description 1M</i>
	(e) Ans.	State three uses of final keyword. Uses of final keyword: 1. Prevent overriding of method: To disallow a method to be overridden final can be used as a modifier at the start of declaration. Methods written as final cannot be overridden. e.g. class test {	4M <i>Prevent overriding :1½M</i>



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>final void disp() //prevents overriding { System.out.println("In superclass"); } } class test1 extends test { voiddisp(){ // ERROR! Can't override System.out.println("Illegal!"); } } } 2. Prevent inheritance: Final can be used to even disallow the inheritance, to do this a class can be defined with final modifier, declaring a class as final declares all its method as final e.g. final class test { void disp() { System.out.println("In superclass"); } } }Class test1 extends test // error as class test is final { ... } </pre> <p>3. Declaring final variable: Variable declared final, it is constant which will not and can not change. final int FILE_NEW = 1;</p>	<p><i>Prevent inheritance :1½M</i></p> <p><i>Declaring final: 1M</i></p>																				
4.	<p>(A) (a) Ans.</p>	<p>Attempt any THREE of the following: Write all primitive data types available in Java with their storage sizes in bytes.</p> <table border="1" style="width: 100%; text-align: center; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">Sr. No.</th> <th style="width: 30%;">Type</th> <th style="width: 20%;">Keyword</th> <th style="width: 40%;">Width (bytes)</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>long integer</td> <td>long</td> <td>8</td> </tr> <tr> <td></td> <td>Short integer</td> <td>short</td> <td>2</td> </tr> <tr> <td></td> <td>Integer</td> <td>int</td> <td>4</td> </tr> <tr> <td></td> <td>Byte</td> <td>byte</td> <td>1</td> </tr> </tbody> </table>	Sr. No.	Type	Keyword	Width (bytes)	1	long integer	long	8		Short integer	short	2		Integer	int	4		Byte	byte	1	<p>3 x 4 =12 4M</p> <p><i>Storage Size of any 4:1M each</i></p>
Sr. No.	Type	Keyword	Width (bytes)																				
1	long integer	long	8																				
	Short integer	short	2																				
	Integer	int	4																				
	Byte	byte	1																				



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

			2	Double Float	double float	8 4		
			3	Character	char	2		
			4	Boolean	Boolean	1 bit		
	<p>(b) Ans.</p>	<p>What is thread priority? Write default priority values and methods to change them.</p> <p>Thread Priority: In java each thread is assigned a priority which affects the order in which it is scheduled for running. Thread priority is used to decide when to switch from one running thread to another. Threads of same priority are given equal treatment by the java scheduler.</p> <p>Default Priority Values: Thread priorities can take value from 1 to 10. Thread class defines default priority constant values as</p> <p>MIN_PRIORITY = 1</p> <p>NORM_PRIORITY = 5 (Default Priority)</p> <p>MAX_PRIORITY = 10</p> <p>1. setPriority: Syntax: <code>public void setPriority(int number);</code> This method is used to assign new priority to the thread.</p> <p>2. getPriority: Syntax: <code>public int getPriority();</code> It obtain the priority of the thread and returns integer value.</p>						<p style="text-align: center;">4M</p> <p style="text-align: center;"><i>Thread Priority explanation 1M</i></p> <p style="text-align: center;"><i>Default priority values 1M</i></p> <p style="text-align: center;"><i>Each method 1M</i></p>
	<p>(c) Ans.</p>	<p>Write a program to generate Fibonacci series 1 1 2 3 5 8 13 21 34 55 89.</p> <pre>class FibonacciSeries { public static void main(String args[]) { int num1 = 1, num2 = 1, ans; System.out.println(num1); while (num2 < 100) { System.out.println(num2); ans = num1 + num2; num1 = num2;</pre>						<p style="text-align: center;">4M</p> <p style="text-align: center;"><i>Syntax 2M</i></p> <p style="text-align: center;"><i>Logic 2M</i></p>



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>num2=ans; } } }</pre>																						
(d) Ans.	<p>Differentiate between Applet and Application (any 4 points).</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">Sr. No.</th> <th style="width: 40%;">Applet</th> <th style="width: 50%;">Application</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td>Applet does not use main() method for initiating execution of code.</td> <td>Application uses main() method for initiating execution of code.</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Applet cannot run independently.</td> <td>Application can run independently.</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Applet cannot read from or write to files in local computer.</td> <td>Application can read from or write to files in local computer.</td> </tr> <tr> <td style="text-align: center;">4</td> <td>Applet cannot communicate with other servers on network.</td> <td>Application can communicate with other servers on network.</td> </tr> <tr> <td style="text-align: center;">5</td> <td>Applet cannot run any program from local computer.</td> <td>Application can run any program from local computer.</td> </tr> <tr> <td style="text-align: center;">6</td> <td>Applet are restricted from using libraries from other language such as C or C++.</td> <td>Application are not restricted from using libraries from other language.</td> </tr> </tbody> </table>		Sr. No.	Applet	Application	1	Applet does not use main() method for initiating execution of code.	Application uses main() method for initiating execution of code.	2	Applet cannot run independently.	Application can run independently.	3	Applet cannot read from or write to files in local computer.	Application can read from or write to files in local computer.	4	Applet cannot communicate with other servers on network.	Application can communicate with other servers on network.	5	Applet cannot run any program from local computer.	Application can run any program from local computer.	6	Applet are restricted from using libraries from other language such as C or C++.	Application are not restricted from using libraries from other language.	<p>4M</p> <p><i>Any 4 points 1M each</i></p>
Sr. No.	Applet	Application																						
1	Applet does not use main() method for initiating execution of code.	Application uses main() method for initiating execution of code.																						
2	Applet cannot run independently.	Application can run independently.																						
3	Applet cannot read from or write to files in local computer.	Application can read from or write to files in local computer.																						
4	Applet cannot communicate with other servers on network.	Application can communicate with other servers on network.																						
5	Applet cannot run any program from local computer.	Application can run any program from local computer.																						
6	Applet are restricted from using libraries from other language such as C or C++.	Application are not restricted from using libraries from other language.																						
4.	(B) (a) Ans.	<p>Attempt any ONE of the following:</p> <p>Write a program to draw a bar chart for plotting students passing percentage in last 5 years. <i>(Note: Any other logic can be considered)(HTML file with separate applet tag may also be considered)</i></p> <pre>import java.awt.*; import java.applet.*; /* <Applet code="BarChart" width=400 height=400> <param name="columns" value="5"></pre>	<p>1 x 6 = 6 6M</p> <p><i>Applet tag</i> 2M</p>																					



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<pre><param name="c1" value="80"> <param name="c2" value="90"> <param name="c3" value="100"> <param name="c4" value="85"> <param name="c5" value="95"> <param name="label1" value="2012"> <param name="label2" value="2013"> <param name="label3" value="2014"> <param name="label4" value="2015"> <param name="label5" value="2016"> </Applet> */ public class BarChart extends Applet { int n=0; String label[]; int value[]; public void init() { try { n=Integer.parseInt(getParameter("columns")); label=new String[n]; value=new int[n]; label[0]=getParameter("label1"); label[1]=getParameter("label2"); label[2]=getParameter("label3"); label[3]=getParameter("label4"); label[4]=getParameter("label5"); value[0]=Integer.parseInt(getParameter("c1")); value[1]=Integer.parseInt(getParameter("c2")); value[2]=Integer.parseInt(getParameter("c3")); value[3]=Integer.parseInt(getParameter("c4")); value[4]=Integer.parseInt(getParameter("c5")); } catch(NumberFormatException e) {</pre>	<p><i>Syntax</i> 2M</p> <p><i>Logic</i> 2M</p>
--	--	--



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>System.out.println(e); } } public void paint(Graphics g) { for(int i=0;i<n;i++) { g.setColor(Color.red); g.drawString(label[i],20,i*50+30); g.setColor(Color.green); g.fillRect(50,i*50+10,value[i],30); } } }</pre>	
	(b) Ans.	<p>What is garbage collection in Java? Explain finalize method in Java. (Note: Example optional) Garbage collection:</p> <ul style="list-style-type: none">• Garbage collection is a process in which the memory allocated to objects, which are no longer in use can be freed for further use.• Garbage collector runs either synchronously when system is out of memory or asynchronously when system is idle.• In Java it is performed automatically. So it provides better memory management.• A garbage collector can be invoked explicitly by writing statement System.gc(); //will call garbage collector. <p>Example:</p> <pre>public class A { int p; A () { p = 0; } } class Test { public static void main(String args[])</pre>	6M <i>Garbage collection explanation 4M</i>



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>{ A a1= new A(); A a2= new A(); a1=a2; // it deallocates the memory of object a1 }</pre> <p>Method used for Garbage Collection finalize:</p> <ul style="list-style-type: none">• The java.lang.Object.finalize() is called by garbage collector on an object when garbage collection determines that there are no more reference to the object.• A subclass override the finalize method to dispose of system resources or to perform other cleanup.• Inside the finalize() method, the actions that are to be performed before an object is to be destroyed, can be defined. Before an object is freed, the java run-time calls the finalize() method on the object. <p>General Form :</p> <pre>protected void finalize() { // finalization code here }</pre>	<p><i>Finalize method explanation on 2M</i></p>
5.	(a) Ans.	<p>Attempt any TWO of the following:</p> <p>What is exception? WAP to accept a password from the user and throw “Authentication Failure” exception if the password is incorrect.</p> <p>An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program execution. It can be handled by 5 keywords in java as follows :</p> <ol style="list-style-type: none">1) try: This block monitors the code for errors.2) catch: This block implements the code if exception is raised due to some error in try block.3) throw: To throw a user define exception4) throws: Can be used with the method’s declaration which are may have some run time errors.5) finally: Includes the code which executes irrespective of errors in try block. <p>Program :</p> <pre>import java.io.*;</pre>	<p>2 x 8 = 16 8M</p> <p><i>Exception 2M</i></p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>class PasswordException extends Exception { PasswordException(String msg) { super(msg); } } class PassCheck { public static void main(String args[]) { BufferedReader bin=new BufferedReader(new InputStreamReader(System.in)); try { System.out.println("Enter Password : "); if(bin.readLine().equals("abc123")) { System.out.println("Authenticated "); } else { throw new PasswordException("Authentication failure"); } } catch(PasswordException e) { System.out.println(e); } catch(IOException e) { System.out.println(e); } } }</pre>	<p><i>Correct logic 3M</i></p> <p><i>Correct Syntax 3M</i></p>
	(b) Ans.	<p>Write a program to create two threads, one to print numbers in original order and other in reverse order from 1 to 10.</p> <pre>class thread1 extends Thread {</pre>	8M



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>public void run() { for(int i=1 ; i<=10;i++) { System.out.println("Thread 1 :" +i); } } } class thread2 extends Thread { public void run() { for(int i=10 ; i>=1;i--) { System.out.println("Thread 2 :" +i); } } } class test { public static void main(String args[]) { thread1 t1 = new thread1(); thread2 t2= new thread2(); t1.start(); t2.start(); } }</pre>	<p><i>Correct Logic 4M</i></p> <p><i>Correct syntax 4M</i></p>
(c)	<p>Explain the following methods of applet class:</p> <p>(i) drawRect()</p> <p>(ii) drawPolygon()</p> <p>(iii) drawArc()</p> <p>(iv) drawRoundRect()</p>		<p>8M</p>
Ans.	<p>(i) drawRect():</p> <p>The drawRect() method displays an outlined rectangle. <i>Syntax:</i> void drawRect(inttop, intleft, intwidth, int height) The upper-left corner of the Rectangle is at top and left. The dimension of the Rectangle is specified by width and height. <i>Example:</i></p>		<p><i>Each method 2M</i></p>



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<p>g.drawRect(10,10,60,50);</p> <p>(ii) drawPolygon(): drawPolygon() method is used to draw arbitrarily shaped figures. <i>Syntax:</i> void drawPolygon(int x[], int y[], int numPoints) The polygon's end points are specified by the co-ordinates pairs contained within the x and y arrays. The number of points define by x and y is specified by numPoints. <i>Example:</i> int xpoints[]={ 30,200,30,200,30}; int ypoints[]={ 30,30,200,200,30}; int num=5; g.drawPolygon(xpoints,ypoints,num);</p> <p>(iii) drawArc(): It is used to draw arc . <i>Syntax:</i> void drawArc(int x, int y, int w, int h, int start_angle, int sweep_angle); where x, y starting point, w & h are width and height of arc, and start_angle is starting angle of arc sweep_angle is degree around the arc <i>Example:</i> g.drawArc(10, 10, 30, 40, 40, 90);</p> <p>(iv) drawRoundRect(): It is used to draw rectangle with rounded corners. <i>Syntax :</i> drawRoundRect(int x,int y,int width,int height,int arcWidth,int arcHeight) Where x and y are the starting coordinates, with width and height as the width and height of rectangle. arcWidth and arcHeight defines by what angle the corners of rectangle are rounded. <i>Example:</i> g.drawRoundRect(25, 50, 100, 100, 25, 50);</p>	
--	---	--

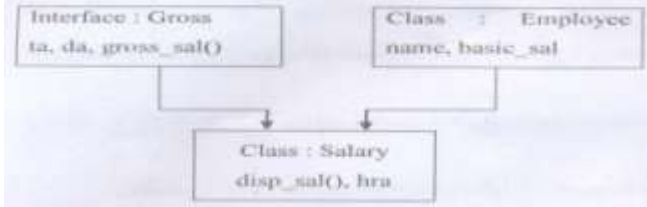


MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

6.	(a)	<p>Attempt any FOUR of the following: Write a program to implement following inheritance:</p>  <pre>classDiagram class Gross { ta da gross_sal() } class Employee { name basic_sal } class Salary { disp_sal() hra } Gross < -- Employee Employee < -- Salary</pre> <p>Ans.</p> <pre>interface gross { int ta=1000; int da=4000; public void gross_sal(); } class employee { String name="Abc"; int basic_sal=8000; } class salary extends employee implements gross { int hra; int total=0; salary(int h) { hra=h; } public void gross_sal() { total=basic_sal+ta+da+hra; } void disp_sal() { gross_sal(); System.out.println("Name :"+name); System.out.println("Total salary :"+total); }</pre>	<p>4 x 4 = 16 4M</p> <p><i>Correct Logic 2M</i></p> <p><i>Correct syntax 2M</i></p>
----	-----	---	---



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		<pre>public static void main(String args[]) { salary s = new salary(3000); s.disp_sal(); } }</pre>	
(b)	What is the use of ArrayList class? State any two methods with their use from ArrayList.		4M
Ans.	Use of ArrayList class: 1. ArrayList supports dynamic arrays that can grow as needed. 2. ArrayList is a variable-length array of object references. That is, an ArrayList can dynamically increase or decrease in size. ArrayLists are created with an initial size. When this size is exceeded, the collection is automatically enlarged. When objects are removed, the array may be shrunk. Methods of ArrayList class : 1. void add(int index, Object element) Inserts the specified element at the specified position index in this list. Throws IndexOutOfBoundsException if the specified index is out of range (index < 0 index >size()). 2. boolean add(Object o) Appends the specified element to the end of this list. boolean addAll(Collection c) Appends all of the elements in the specified collection to the end of this list, in the order that they are returned by the specified collection's iterator. Throws NullPointerException if the specified collection is null. 3. boolean addAll(int index, Collection c) Inserts all of the elements in the specified collection into this list, starting at the specified position. Throws NullPointerException if the specified collection is null. 4. void clear() Removes all of the elements from this list. 6. Object clone() Returns a shallow copy of this ArrayList.	<i>Use 2M</i> <i>Any two methods 2M</i>	



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	<p>5. boolean contains(Object o) Returns true if this list contains the specified element. More formally, returns true if and only if this list contains at least one element e such that (o==null ? e==null : o.equals(e)).</p> <p>6. void ensureCapacity(intminCapacity) Increases the capacity of this ArrayList instance, if necessary, to ensure that it can hold at least the number of elements specified by the minimum capacity argument.</p> <p>7. Object get(int index) Returns the element at the specified position in this list. Throws IndexOutOfBoundsException if the specified index is out of range (index < 0 index >= size()).</p> <p>8. intindexOf(Object o) Returns the index in this list of the first occurrence of the specified element, or -1 if the List does not contain this element.</p> <p>9. intlastIndexOf(Object o) Returns the index in this list of the last occurrence of the specified element, or -1 if the list does not contain this element.</p> <p>10. Object remove(int index) Removes the element at the specified position in this list. Throws IndexOutOfBoundsException if index out of range (index < 0 index >= size()).</p> <p>11. protected void removeRange(intfromIndex, inttoIndex) Removes from this List all of the elements whose index is between fromIndex, inclusive and toIndex, exclusive.</p> <p>12. Object set(int index, Object element) Replaces the element at the specified position in this list with the specified element. Throws IndexOutOfBoundsException if the specified index is out of range (index < 0 index >= size()).</p> <p>13. int size()</p>	
--	--	--



MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

		Returns the number of elements in this list. 14. Object[] toArray() Returns an array containing all of the elements in this list in the correct order. Throws NullPointerException if the specified array is null. 15. Object[] toArray(Object[] a) Returns an array containing all of the elements in this list in the correct order; the runtime type of the returned array is that of the specified array. 16. void trimToSize() Trims the capacity of this ArrayList instance to be the list's current size.	
(c) Ans.	Design an applet which accepts username as a parameter for html page and display number of characters from it. import java.awt.*; import java.applet.*; public class myapplet extends Applet { String str=""; public void init() { str=getParameter("uname"); } public void paint(Graphics g) { int n= str.length(); String s="Number of chars = "+Integer.toString(n); g.drawString(s,100,100); } } /*<applet code=myapplet height=200 width=200> <param name="uname" value="student1"> </applet>*/	4M <i>Correct Logic 2M</i> <i>Correct syntax 2M</i>	
(d)	List any four built-in packages from Java API along with their use.	4M	



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Java Programming

Subject Code: 17515

	Ans.	<ol style="list-style-type: none">1. java.lang: Contains Language support classes which are used by Java compiler during compilation of program2. java.util: Contains language utility classes such as vectors, hash tables, random numbers, date etc.3. java.io: Contains I/O support classes which provide facility for input and output of data.4. java.awt: Contains a set of classes for implementing graphical user interface.5. java.applet: Contains classes for creating and implementing applets.6. java.sql: Contains classes for database connectivity.7. java.net: Contains classes for networking.	<i>Any 4 packages with its use 1M each</i>
	(e) Ans.	Write a program to accept two numbers as command line arguments and print the addition of those numbers. class addition { public static void main(String args[]) { int a,b; a= Integer.parseInt(args[0]); b=Integer.parseInt(args[1]); int c= a+b; System.out.println("Addition= "+c); } }	4M <i>Correct Logic 2M</i> <i>Correct syntax 2M</i>