**Question 1**

What is the output For the program

**public** **class** Number{

**public** **static** **void** main(String args[]) {

String 2010String = "Year 2010 ";

String $dollarStr = 2010String + "Dollar Value!!";

System.out.println($dollarStr);

}

}

Solution: Will not compile as variables cannot start with numbers

**Question 2**

Explain Final and Abstract Classes?

Solution: Abstract classes have to be extended to be used where as a final class cannot be sub-classed.

**Question 3**

What is the importance of hashCode() and equals() methods? How are they used in java?

Solution: These two methods are implemented in java.lang.object.

The hashcode() method provided by the object is derived by mapping the memory address to an integer value. It is the native implementation which provides the memory address to a certain extent.

The equal’s method is used to make comparison between two objects. It can be overridden to implement a custom method of implementation.

**Question 4**

Explain java heap vs stack memory? What is the role of garbage collector with these?

Solution: The stack memory holds local variable and partial results and plays a part in method invocation and return.

Heap memory is the runtime data area from which memory for all class instances and arrays is allocated. Heap storage for objects in reclaimed by an automatic storage management system (known as a garbage collector); objects are never explicitly deallocated.

**Question 5**

What is the difference between prepared statement and callable statement?

Solution:

**Question 6**

What is the difference between Serializable and Externalizable interface?

Solution: Serializable is a tagging interface; It serves to assign the serializable data types to the tagged class and to identify the class as one which has been designed for persistence.

Externalizable: When the serialization process is to be controlled then Externalizable interface is used. The externizable interface extends serializable and adds two methods, write external and readexternal which are automatically called during serialization and deserialization.

**Question 7**

Which method names follow the JavaBeans standard?

a. addSum

b. getMail

c. delMail

e. isEmail

Solution: getMail, isEmail

**Question 8**

Write an efficient program to split a comma separated string?

Solution:

public class StringSplit

{

public static void main(String[] args)

{

String data = "1,Diego Maradona,Footballer,Argentina";

String[] items = data.split(",");

for (String item : items)

{

System.out.println("item = " + item);

}

}

}

**Question 9**

What is the difference between Boolean & operator and the && operator?

Solution: The && and || are short circuit operators. A short circuit operator is one that doesn’t evaluate all of its operands.

The & operator which is a Boolean operator evaluates both the expressions.

**Question 10**

In the program given below will the isOdd() method be able to identify the even and odd numbers correctly?

**public** **class** test {

**public** **static** **void** main(String[] args) {

**for**(**int** i=-2;i<=2;i++) {

**if**(*isOdd*(i)){

System.*out*.println("Odd:"+i);

} **else** {

System.*out*.println("Even:"+i);

}

}

}

**public** **static** **boolean** isOdd(**int** i) {

**return** i % 2 == 1;

}

}

Solution: No

**Question 11**

Explain Servlet life cycle?

Solution:

Call init(): servlet will be initialized using web.xml

Call service(): this will be called upon each request which in turn calls doXXX() methods.

call destroy(): called when the servlet object is no longer needed.

**Question 12**

How is a session maintained in a web application?

Solution: Applications with EJB use stateful session beans, using httpSession, hidden form fields, url rewriting, cookies

**Question 13**

What is a RequestDispatcher? What object do you use to forward a request?

Solution: RequestDispatcher object can forward a clients request to a resource or include itself in the response back to the client. A resource can be another servlet, or an HTML file, or a JSP file etc.

To illustrate, suppose you want Servlet\_A to invoke Servlet\_B. If they are both in the same directory, you could accomplish this by incorporating the following code fragment in either the service method or the doGet method of Servlet\_A:

RequestDispatcher dispatcher = getRequestDispatcher("Servlet\_B");

dispatcher.forward( request, response );

**Question 14**

Write program to reverse a linked list?

Solution:

Logic for reverse.

public void reverse()  
{  
 Node currentNode, nextNode, loopNode;  
 if(first==null)  
 return;  
  
 currentNode=first;  
 nextNode= first.next;  
 loopNode=null;  
  
  
 while(nextNode != null)  
 {  
 currentNode.next = loopNode;  
 loopNode= currentNode;  
 currentNode=nextNode;  
 nextNode =nextNode.next;  
 }  
  
 first = currentNode;  
 first.next = loopNode;  
  
}

**Question 15**

Write a program/pseudocode to print a histogram of the frequencies of different characters in its input.

**Question 16**

Write the pseudocode for binary search?

Solution:

Algorithm binarySearch(list[], searchTarget)

last <-- length(list[])

first <-- 1

// while there are still elements to be searched through

while first = last do

middle <-- (first + last) / 2

// if current middle value is the search target

if list[middle] = searchTarget

return middle

// if current middle value is less than the search target

else if list[middle] < searchTarget

first <-- middle + 1

// if current middle value is larger than the search target

else

last <-- middle - 1

end if

end while

// return 0 if search target not found

return 0

**Question 17**

Examine the data and answer the following questions

**EMPLOYEES**

|  |  |  |
| --- | --- | --- |
| **LAST\_NAME** | **DEPARTMENT\_ID** | **SALARY** |
| Getz | 10 | 3000 |
| Davis | 20 | 1500 |
| Bill | 20 | 2200 |
| Davis | 30 | 5000 |
| Kochhar |  | 5000 |

**DEPARTMENTS**

|  |  |
| --- | --- |
| **DEPARTMENT\_ID** | **DEPARTMENT\_NAME** |
| 10 | Sales |
| 20 | Marketing |
| 30 | Accounts |
| 40 | Adminstration |

Retrieve all employees, whether or not they have matching departments in the department table.

Solution: SELECT last\_name, department\_name from employees e LEFT OUTER JOIN departments ON (e.department\_id=d.department\_id);

**Question 18**

Describe Action classes and Action forms?

An action class is the struts application extends struts class. Action class acts as wrapper around the business logic and provides an interface to the applications model layer. It acts as glue between view and model layer. It also transfers data from the view layer to the specific business process layer and finally returns the processed data from the business layer to the view.

**Question 19**

What is the difference between schema and DTD?

Solution:

**A DTD is:**

The XML Document Type Declaration contains or points to markup declarations that provide a grammar for a class of documents. This grammar is known as a document type definition or DTD.

The DTD can point to an external subset containing markup declarations, or can contain the markup declarations directly in an internal subset, or can even do both.

**A Schema is:**

XML Schemas express shared vocabularies and allow machines to carry out rules made by people. They provide a means for defining the structure, content and semantics of XML documents

**Question 20**

Analyze the following data

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EMPLOYEE\_ID | EMP\_NAME | DEPT\_ID | MGR\_ID | JOB\_ID | SALARY |
| 101 | Smith | 20 | 120 | SA\_REP | 4000 |
| 102 | Martin | 10 | 105 | CLERK | 2500 |
| 103 | Chris | 20 | 120 | IT\_ADMIN | 4200 |
| 104 | John | 30 | 108 | HR\_CLERK | 2500 |
| 105 | Diana | 30 | 108 | HR\_MGR | 5000 |
| 106 | Bryan | 40 | 110 | AD\_ASST | 3000 |
| 108 | Jennifer | 30 | 110 | HR\_DIR | 6500 |
| 110 | Bob | 40 |  | EX\_DIR | 8000 |
| 120 | Ravi | 20 | 110 | SA\_DIR | 6500 |

Write a query to get the list of ID, Name and salary of the employee, and the ID and name of the employee’s manager, for all the employees who have a manager and earn more than 4000.

Solution: Select e.employee\_id “Emp\_id”, e.emp\_name “Employee”, e.salary, m.employee\_id “Mgr\_id”, m.emp\_name “Manager” FROM employees e, employee m

WHERE e.mgr\_id =m.employee\_id and e.salary> 4000