① → Public enum LeaveType{
EL;
} → Value of enum?

① enum LeaveType.EL
② LeaveType.EL
③ EL
④ enum.EL

③ ~~8~~ Public static void main
~~int~~ n=2.
If (n=2)
SOP.(" Welcome")
else SOP("hi");

a) prints welcome
b) runtime error
c) compilation error at line 2
c) prints hii.

④ Bike riding paragarph
a) Polymorphism
✓ b) Abstraction
c) Encapsulation
d) Package

⑦

② SOP ("13" +5 +3);
a) It will give a runtime error
b) 1353 ✓

⑤ Switch case.
number = 30;
Case 10 ~~10~~;
Case 20;
Case 30;
default SOP(Not in 10,20,30);

a) Compilation error
b) 10,20,30, ; not in 10,20,30
c) 30
d) 30; not in 10,20,30

6) String S1 = "abc"
String S2 = "abc")
SOP("S1 == S2 is : " + S1==S2);

a) S1 == S2 is true.
b) false
c) true
d) S1 == S2 is : false

⑦ .... void method (int a) { }
True?

a) This is an incorrect method.
~~b) This method takes int array as argument.~~
~~c) This method can only invoke.~~
d) This method returns an int.

⑧ Can an object of a child type be assiged to a variable of the Parent type ?

```
// Card crd; // Card class
Birthday bd = new BirthDay ("Lucinda, 42);
crd = bd
```

a) No - but a object of Parent type can be assigned

b) Yes - an object can be assigned to a reference.

c) No, - there must always be an exact ... types.

d) Yes - any object can be assiged to any

⑨
```
Public Class Student extends String {
    Public void fun () {
            SOP ("String in immutable");
    } };
```

Output ?

a) will not compile because String is abstract

b) " " " " Student class

c) " " " " the class is not

d) " " " " String class method

10)
```
Public class pro? {
    int a = 18;
  PSV. call () {
       SOP ("Age" +a);
  }
     PSVM () {
      Pro ob = newpro();
       ob. call ();
     }
}
```

a) 23

b) Cannot make a static referenc to the.

c) 18

d) 24.

11)
```
Public class Main
   {
     P S VM (String args[]) {
          SOP (fun());
      }
      int fun() {
          return 20;
        }
        }
```

a) 20

b) 0

c) Garbage value

d) Compiler error: Non static method cannot

12)
```
Public class Strings{
     P SVM (String ads[]) {
    String arr[] = {{"meow", "bray", "moo"};
     String a = "meow";
       SOP (arr[0] = = a);
      }
     }
```

a) Compiler error : at line 5

b) meow

c) true d) false.

13)
```
Public class MyDemo
     static int count = 0;
     MyDemo()
        { Count++; }
     PS v disp()
        { SOP (Count) ; }
     PS vm (String args[])
     { MyDemo obj1 = new My Demo();
       MyDemo obj2 =  "   "   "
         obj1.disp()
        }
```

a) 1

b) 2

c) Error : Static methods cannot be called using ...

d) 0

Scanned by CamScanner

14) Which of the following statement is not object creation statement for class student?

a) Student S1= new ~~stu~~ Student(), S2 - new Student()

b) Student S1= new Student();

c) new Student();

d) Student S1;

15)
```
① Public class EnumString {
② Public enum TextStyle {
③ ~~Bold~~ BOLD, ITALICS, UNDERLINE, STRIKETHROUGH.
   }
   PSVM (String[] args) {
   String style = "Bold");
   Textstyle textstyle = Text Style .value Of (style.toUpperCase());
     SOP (textStyle);
```

a) BOLD

b) Same as line ③

c) bold

d) Compilation Error.

16.
```
Public Class test {
  Private Static float temp()
   { static float sum =21;
       return (--(sum));
   } PSVM { Test test = new Test();
       SOP (test.temp());
     }
```

a) 21

b) Compilation error illegal modifier (static) for sum

c) 20

d) Runtime error.

17) Which of the following features of object orientation enables an object to behave under different circumstances?

a) ans Polymorphism.

18)
```
class Test
 {
 PSVM (String [] args)}
  {
int temp = 0;
~~char~~ data = null;
    SOP (temp + " " + data);
```

a) Runtime error.

b) Compilation error due to data

c) 0 null.

d) null null

19)
```
Public Class Weather {
    Static boolean isRaining;
    PSVM (String args[]);
     {  SOP (isRaining);
       } };
```

a) Does not compile as boolean is not initialized

b) None.

c) Prints false.

d) Prints true.

20)
```
P.S VM (Strings args[]) {
    new string a = "abc"
      new string b = "abc"
        SOP( a.equals (b));
        SOP (a = = b);
```

a) True, True

b) False, False

c) False, true

d) true, False.

21) Name obj = null; ? ← Null Point

22) Class check {

```
  PSVM (String[] args) {
  {
     throw new Hull Pointer Exception ("Hello");
  {
   Catch (Airthmatic Exception e) {
      Sop ("B");
```

a) Hello

b) Runtime error

c) B

d) Compilation error

23) Multiple Catch Block 1 {

```
    PS VM (String args []) {
       {
  int a[] = new int [5];
    a[5] = 30/0;
       }
      Catch (Exception e) {Sop ("Common Task Compleated");}
        Catch (Airthmetic Exceptione) { SOP ("task1 is completed")
  Catch (Array Index Out of Bounds Exception e) {SOP ("task 2
   completed");}
            SOP ("rest of the code...");
       }
     }
```

a) Common task completed

b) Compilation error: Airthmetic Exception and already been caught.

c) task 1 is completed.

d) task 2 is completed.

24) which one of the following Keywords is used to specify a list of exceptions that a method may raise?.

a) throws

b) extends

c) throw

d) implements.

25)
```
.... class Student {
    .... void check (int experience)
    {
        (... perience > 2)
        ... new CheckException (" Cannot apply");
        { SOP(" ~~eta~~ il eligible to apply"); }
    } ~~fo~~s PSVM (String args[])
       . . .

          ..(2);
      }
```

28)

```
Class Super {
 public int i=0;
 Super () {}
  ~~Pa Sut~~ Public Super (String S1) {
   i = 1;
  } }
 Public Class Sub extends Super {
   Public Sub (String S1) {
     i = 2;
   }
   P S VM (String [] args) {
    Sub sub = new Sub (" Hello");
     SOP (Sub. i); }}
```

a) 0

b) 2

c) 1

d) None of thes.

29)

```
 import java.util. HashMap;
  "       "      "   Map.
  Public class MapTest {
    P S VM (String aga[]) {
      Map m = new HashMap();
       m.put (null, "Test");
        m. put (null, "Fest");
          SOP (m);
          }
```

a) Null pointer Exception

b) Compilation error

c) {null = Test}

d) null = Fest }

```
(30) import java. util. HashSet;
class Animal{
    Public int i=12;
      Pbl public Animal(){
          i=13;
          }
       Public String toString(){
         return "Animal"+i;
          }
          }
       Public class Test {
       PSVM (String[] args) throws Exception {
         HashSet <Animal> s= new HashSet <Animal>();
          S. add(new Animal());
           s. add(new Animal());
            for (Animal a : S) {
              SOP(a)
              }
              }
```

a) Animal 13

b) Animal 13
   Animal 13

c) Runtime error about duplicate element

d) Animal 0

(31)
```
Class Employee {
  Public Static Employee [] listAll() {
    List <Employee> es = dao().list();
    return es.toArray(new Employee [es.size()]);
    }
  }
```

In the above snippet, which of the following is the Correct way method of Employee class in static way?

a) String emp = Employee.listAll().
b) Employee employee = Employee. list All();
c) Employee [] employee = Employee. listAll();
d) Employee e = new Employee(); e.listAll();

(32)
```
Class Account {
  }
  Class BankAccount extends Account {
      int a = 22;
        private void getAccountNo (int a) {
                SOP (a);
          } };
    Public class AccountTest
     {
          ~~Public s~~ PSVM (String [] args) {
          BankAccount c= new BankAccount();
           c.getAccountNo(12);
          }
          }
```

a) c.getAccountNo() is not allowed as it is defined with private access specifier.

b) Private access Specifier is not allowed in Bankaccount class

c) 12

d) 22

(33) Which of follwing statement is true for inheritance in java?

a) Java extend multiple classes.

b) Extends Keyword is used to inherit both class and interface.

c) interface extends multiple interfaces in java.

d) final classes can be extended in java.

c) Prints " Test Class : tests"

d) Code complains : override method tests() is static at line 13 .