***Practical 4: Questions***

Q1. What is the signature for methods in Java?

Return\_type method\_name (parameters){

//body

}

Q2. What are the different access specifiers for methods?

Private, public, protected, default

Q3. How to invoke any class method in a java program?

By making an object of that class and then calling that method

Ex: class\_name obj1 = new class\_name(Parameters for constructors if any);

Obj1.method\_name(parameters if any)

Q4. What is method overloading in Java?

Having the same name but different number of parameters or other data types of parameters or further order of parameters

Q5. What is a constructor?/ How constructors are different from methods/ What are the rules to define a constructor?

It is a special member function that does not have a return data type. Its name is the same as that of the class name. It gets invoked automatically when the object of the class is created. It gets invoked only when the object is created and it cannot be called ‘n’ number of times like in the case of methods.

Q6. What are the different types of constructors in JAVA? Explain them

Default constructor and parameterized constructor

Class A{

Int x;

A(){ // Default constructor:

x = 0;

}

A(int x){ // Parameterized constructor

this.x = x;

}

}

Q7. Can we give access modifiers to constructors?

All 4 access modifiers are allowed to the constructor. We should declare the constructor as private for not to allow the user to create objects from outside our class. We will declare a private constructor in the Singleton design pattern

Q8. Can we make the constructor abstract, final, or static?

In Java, a constructor is not allowed to be abstract, final, static, native, or strictfp. So, there is no static constructor in Java. A static constructor used to initialize static data means the specified task will execute only once throughout the program.