Assignment 1

**1. What is SDLC?**

* SDLC means Software Development Life Cycle
* SDLC is used by software industry to design, to develop application
* SDLC process start from client requirement to submit whole application to client.
* SDLC has 6 phase

A) Requirement analysis and information gathering

B) Designing

C) Development

D) Testing

E) Deployment

F) Maintenance

**2. What is software testing?**

- Process of checking your application or website working properly and fulfil all requirements of client.

Types of software testing

1 Unit testing

-unit testing is check module at a time and to check all included components and functionality working properly.

2 Integrated testing

Integrated testing means to test more than one module and and check all information sent by one module is received by another module properly. Each module connected with each other are working properly

3 System testing

It is final and complete testing beforehand submitting application to client. System testing check whole application or website is working properly, requirements of client is well included

This testing is final testing by developer side

4 UAT user acceptance testing

This testing conducted by client or user

**3. What is OOPS ?**

- It is stands for Object oriented programming Language.

- The purpose of OOPS is to deal with real time entity using programming language.

- Benefits of OOPS

1-Code reusability

2-Easier Maintenance

3-Improved readability

4-Scalability

**4. Write Basic concept of OOPS?**

1. Class :- Class is collection of object and method.

Its doesn’t need any memory or space.

Syntax:- access\_modfier class\_keyword class\_name

2. Object :- object is instance of class that executes class.

3. Encapsulation :- It wraps data.

4. Inheritance :- To use data from inherited class by using extends keyword for code usability

5. Polymorphism:- one interface multiple implementation

6. Abstraction:- To hide sensitive data in abstract class and abstract method.

**What is object?**

-An object in object-oriented programming represents a real-world entity by grouping its characteristics (**attributes**) and behaviors (**methods**) together, making it easier to manage and use in a program.

-It acts as a container that **holds non-primitive data types**, generally made up of classes, arrays, and interfaces.

-The syntax to create an object is: *class\_name object\_name = new class\_name();*

*Class Name:* The template/Blueprint used to create the object.

*Object Reference:* The named variable that references the object in memory.

*new keyword:* dynamically allocates the necessary memory based on the logic defined in the class constructor.

In summary, *an object is a concrete instance of a class, dynamically allocated in memory, and used to store and manipulate data defined by that class.*

**6.What is class?**

-It is a collection of object and it doesn’t take any space or memory

-A class is a template that consists of a collection of objects, and it itself does not occupy any space or memory.

-It serves as a template for creating objects

-There are two types of classes:

Predefined Classes: These are built-in classes provided by the programming language itself or its standard libraries. Examples include: **String**, **ArrayList**, **HashMap**, **Scanner, System**

User-Defined Classes: These are custom classes defined by the programmer to suit their specific needs. Examples include:

**Car, Book, Bank Account, Person**.

* Class Syntax: The syntax for defining a class is:

*AccessModifier class ClassName*

*{*

*// Data types, variables, constructors, methods*

*} // Class scope*

**7. What is encapsulation?**

-Encapsulation is method or mechanism where we can wraps data, data members and methods of class in a single unit which is called encapsulation.

-Flow of Encapsulation

Class 🡪 method 🡪 Data

-Encapsulation achieved by simple class modifier like private, public and protected.

-Encapsulation has one more type which is advanced encapsulation

- Advanced encapsulation can be used with pojo(plain old java object) class

-POJO class is MVC class which does not contain main method.

-POJO class use getter and setter method to wrap data.

**8.What is inheritance?**

-Inheritance: A child class utilizes the functionality of a parent class by using the extends keyword.

-Access Restrictions: Private members of a class cannot be accessed through inheritance.

-Advantages: Inheritance allows for code reusability and code optimization.

**Types of inheritance**

* **Single: -** Single child and single parent.
* **Multiple: -** Single child and multiple parent. (it does not support java)
* **Multi-level: -** In multi-level inheritance we have only one super class and multiple sub classes is called multi-level inheritance.
* **Hierarchical: -** Single parent multiple child.
* **Hybrid: -** Combination of two inheritances.

**9.What is polymorphism?**

-The term polymorphism comes from "poly" meaning many, and "morphism" meaning forms. It signifies the same object having different behaviours

-It involves a single interface with multiple implementations.

-Types of Polymorphism :

1. Method overriding: The same method name with the same arguments in a subclass.

e.g.

class MathUtils {

int **add**(int a, int b) { return a + b; }

int **add**(int a, int b, int c) { return a + b + c; } // Overloading the add method

}

1. Method overloading: The same method name with different arguments within the same class.

e.g.

*// In Parent Class*

class Animal {

void **makeSound**() { System.out.println("Animal makes sound"); }

}

*// In Child Class*

class Dog extends Animal {

@Override

void **makeSound**() { System.out.println("Dog barks"); } // Overriding the makeSound method

}

**10.Write SDLC phases with basic introduction.**

A) Requirement analysis and information gathering:

In this phase, gathering all requirement from client side and then collect all information about all requirement.

B) Designing:

In this phase, mostly we create and architect all flow of software system, assign developer and all required components. This involves designing the system architecture and the detailed internal components including databases, user interface and modules.

C) Development:

This phase basically for write code and develop whole application.

D) Testing:

Ensure the software works as expected and is free of bugs. Conduct various test to identify and fix defect or errors. This phase ensure that software meet with client requirement.

E) Deployment:

Release whole software in real time server. Give software to client. Now software access by end user

F) Maintenance:

Update, fix and improve the software post-deployment. This phase continues for the software lifecycle, ensuring it remains functional and up-to-date.