java

Assignment-4

C. Gnanendra
192211158

1) Simple Inheritance:-

```
class animal {
    string name;
    void makesound() {
system.out.println("Animal makes a
    }                                sound");
}
class dog extends Animal {
    @ override
    void make sound () {
    sy.system.out.println("The dog barks");
  }
}
Public class main {
public static void main(string [] args){
  Dog dog = new Dog();
  dog.make sound();
  }
  }
```

2) Constructer Inheritance

```
Class person{
  String name;
   int age;
person(string name, int age){
   this.name = name;
   this.age = age;
}
}
```

```
class student extends person{string grade
student (string name, int age, string
grade ) {
        super.(name, age);
        this.grade = grade;
    }void display(){
  Student student =new student
("John", 20, "A");
      student display();
    }
}
```

3) **Multilevel Inheritance :-**

```
class vehicle {
   int speed;
  string fuel Type;
vehicle (int speed, string fuel Type){
   This. speed := speed;
   this. fuel type = fuel type;
    }}
class car extends vehicle{
   int number of Doors;
car (int speed, string fuel Type,
int number of Doors){
super(speed, fuel Type);
```

```
}
}
public Main {
 public static void main (String[] args){
Electric can. electric can = new
Electric can (120, "Electric", 4, 85);
electric.display properties();
    }
    }
```

4) Method overriding in Inheritance

```
class shape {
 void draw(){
system.out.println("Drowing a shape")}
  }
}
class circle extends shape{
    @override
   void draw (){
system.out.println ("Drawing a circle");
 }
}
public class main{
public static void main (string [] args){
  shape circle = new circle();
  circle.draw ();
  }
}
```

5) Inheritance and Access Modifiers:-

```
class Employee{
 private string privateField =
    "private";
    protected string
protected Field = "protected";
public string public Field = "public";
}
class Manager extends Employee
 {
  void displayfield(){
System.out.println(private field);
}
}
public class Main {
  public static void main
(string [] args) {
  Manager manager = new
Manager();
Manager . displayfields();
  }
  }
```