Assignment – 5

1. WAP to have Student class has roll, name and Department object should have id and name. Assign and print individual values in main method (Note - Containment using constructor and getter/setter).
2. Create Employee class which has attributes (id, name, salary, dept, mydate) Where dept and mydate is class, dept has (dept_id, dept_name) And mydate has (day, month, year) Display Employee information (Note - Containment using constructor and getter/setter).
3. Create Person class with fields(id,name,job) where job is another class with field(jobid,profile,joiningdate) where joiningdate is another class with fields (day, month, year) Display Person information (Note - Containment using constructor and getter/setter).
4. Create a class with display() method that prints "This is parent class" and its subclass with another view() method that prints "This is child class". Now, create an object for each of the class and call
   1 - method of parent class by object of parent class
   2 - method of child class by object of child class
   3 - method of parent class by object of child class
5. In the above example, declare the method of the parent class as private and then repeat the first two operations (You will get error in the third).
6. Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize length and breadth of the rectangle. Let class 'Square'

inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class as 'super(s,s)'. Print the area and perimeter of a rectangle and a square.

7. Below code is showing compile time error. Can you suggest the corrections?

```
class X
{
   public X(int i)
   {
      System.out.println(1);
   }
}
class Y extends X
{
   public Y()
   {
      System.out.println(2);
   }
}
```

8. WAP to create a class Kid with method readBook() and another method readBook () with 2 parameters. The method readBook here is over-loaded. Create a class BigKid which extends Kid created above and Implement readBook() differently in BigKid class. Here the method readBook() has been over-ridden in the child class BigKid().
9. Create a class Teenager which extends Kid created above and Implement readBook() differently in Teenager class. In main method, declare two variables k1,k2 of type Kid. Create objects of type BigKid and Teenager such that K1 should reference object of class BigKid and K2 should reference object of class Teenager, Call their respective readBook() methods.

10. Create class Student (id, name) then create 2 objects of Student using new keyword. Print 2 objects and see its hashcode is different E.g. tudent@15db9742 and Student@2329742. If you do this Student s1 = new Student (); Student s2 = s1; Now print s1 and s2 see both reference variables are pointing to same location now override the toString () method in student class and see the beautiful output.
11. What will be the output of the following program?

```
class A
{
   String s = "Class A";
}
class B extends A
{
   String s = "Class B";
    Public B()
   {
      System.out.println(super.s);
   }
}
class C extends B
{
   String s = "Class C";
   Public C()
   {
      System.out.println(super.s);
   }
}
public class MainClass
{
   public static void main(String[] args)
   {
      C c = new C();
      System.out.println(c.s);    }     }
```