**44-542 Object Oriented Programming**

enumTypes Key

1. Assume we have created the following enum type variables:

**enum studentEducation {UG, G, PHD};**

Evaluate each of the following expressions:

1. How do you access and create graduate variable and assign G value from the enum, write one line code.

**studentEducation graduate = studentEducation.G;**

1. What is enum name?

**studentEducation**

1. What is the values of studentEducation.UG?

**UG**

1. Write a line of code to print the value of anyone of the enum variable that you created above

**System.out.println(studentEducation.UG);**

1. Create Java Enum named **Education** that holds four values **UG,G,PHD,SPECIAL**

Create a Student class with two attributes **name** and **age**, follow the below directions

**name** is a **String** variable

**age** is an int type

Complete the constructor for the Student class that takes two attributes and for object creation

Write a method **typeOfStudent** that takes an argument of type **Education** named type and prints the result based on the student education type. Printed result should be as follows:

If student is undergraduate, then method should return “Student is Undergraduate and must complete PHD in order to eligible for the position”

If student is graduate, then method should return “Student is graduate and must complete PHD in order to eligible for the position”

If student is doctorate, then method should return “Student is doctorate and eligible for position”

Create three Student objects s1, s2 and s3 for undergraduate, graduate and doctorate respectively one for each student type and invoke the method that you created above. For method invoking use the **Education** enum class values to pass the parameters.

Solution:

**public enum Education {**

**UG,G,PHD,SPECIAL**

**}**

**//Student Class**

**public class Student {**

**private String name;**

**private int age;**

**public Student(String name, int age) {**

**this.name = name;**

**this.age = age;**

**}**

**public void typeOfStudent(Education type) {**

**if (type.equals(Education.UG)) {**

**System.out.println("Student is Undergraduate");**

**} else if (type.equals(Education.G)) {**

**System.out.println("Student is Graduate");**

**} else if (type.equals(Education.PHD)) {**

**System.out.println("Student is doctorate");**

**}**

**}**

**}**

**public static void main(String[] args) {**

**// TODO code application logic here**

**Student s1 = new Student("Student 1", 23);**

**Student s2 = new Student("Student 1", 23);**

**Student s3 = new Student("Student 1", 23);**

**s1.typeOfStudent(Education.UG);**

**s2.typeOfStudent(Education.G);**

**s3.typeOfStudent(Education.PHD);**

**}**

**}**