B. Tech Program Second Year
Course: Object Oriented Programming (Java) LAB
Course code: CS2131

PASSWORD/OTP GENERATOR

A
Project Report
submitted
in partial fulfillment
for the award of the Degree of
Bachelor of Technology
in **the Department of Computer Science and Engineering**

Project Mentor:
Name: Mr. Vivek Kumar
Designation: Assistant Professor

Submitted By:
Abhivandya Abhiram
219301709

Department of Computer Science and Engineering
Manipal University, Jaipur

Session 2022-23

ACKNOWLEDGEMENTS

I would like to use this opportunity to sincerely thank our university, Manipal University Jaipur to give us this marvelous chance to work on such interesting topic and make this project.

We are also grateful to our mentor and our project guide, Mr. Vivek Kumar for the constant guidance, inspiration, and constructive suggestions he gave that proved to be helpful in the making of this project.

I would also like to thank our friends who have helped in the successful completion of this project.

CONTENTS

INTRODUCTION

Generating temporary password is now a requirement on almost every website now-a-days. In case a user forgets the password, system generates a random password adhering to password policy of the company. Following program generates a random password adhering to following conditions −

- It should contain at least one capital case letter.
- It should contain at least one lower-case letter.
- It should contain one of the following special characters: @, $, #, !.
- It should contain numeric values.

Many a times we forget our passwords and we opt for Forget password option and within no time we get a new password at our registered email-ID or phone no. to login our account. And every time we get a different password. Sometimes we access our bank accounts while shopping from an online store or many more ways, to verify our transactions from the bank account, they send us OTP(One Time Password) on our registered phone no. or our email-ID, within no time.

HARDWARE AND SOFTWARE REQUIREMENTS

SOFTWARE USED IN THE DEVELOPMENT OF THIS PROJECT:

- JDK, JVM

SOFTWARE USED IN THE DEVELOPMENT OF THIS PROJECT:

- OS- Windows 7,10
- RAM- 2GB and above

WORKING

```
import java.util.*;
public class otpp
{
    static char[] OTP(int len)
    {
        System.out.println("Generating OTP using random() : ");
        System.out.print("You OTP is : ");

        // Using numeric values
        String numbers = "0123456789";

        // Using random method
        Random rndm_method = new Random();

        char[] otp = new char[len];

        for (int i = 0; i < len; i++)
        {
            // Use of charAt() method : to get character value
            // Use of nextInt() as it is scanning the value as int
            otp[i] =
             numbers.charAt(rndm_method.nextInt(numbers.length()));
        }
        return otp;
    }

}
public class NewClass extends otpp
{

    public static void main(String[] args)
    {
        // Length of your password as I have choose
        // here to be 8
        int length = 8;
        System.out.println(Password(length));
        int length1 = 5;
        System.out.println(OTP(length1));
    }
    // This is our Password generating method
    // We have used static here, so that we do not have to
    // make any object for it
    static char[] Password(int len)
    {
        System.out.println("Generating password using random() : ");
        System.out.print("Your new password is : ");
```

```
        // A strong password has Cap_chars, Lower_chars,
        // numeric value and symbols. So we are using all of
        // them to generate our password
        String Capital_chars = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
        String Small_chars = "abcdefghijklmnopqrstuvwxyz";
        String numbers = "0123456789";
                String symbols = "!@#$%^&*_=+-/.?<>)";


        String values = Capital_chars + Small_chars +
                        numbers + symbols;

        // Using random method
        Random rndm_method = new Random();
        char[] password = new char[len];
        for (int i = 0; i < len; i++)
        {
            // Use of charAt() method : to get character value
            // Use of nextInt() as it is scanning the value as int
            password[i] =
              values.charAt(rndm_method.nextInt(values.length()));

        }
        return password;
    }
}
```

Output:

1. Password length- 10, OTP length-4

```
Generating password using random() :
Your new password is : 1/G5Ht=XQj
Generating OTP using random() :
You OTP is : 7420
```

---

2. Password length- 8, OTP length-5

```
Generating password using random() :
Your new password is : cKQATPmy
Generating OTP using random() :
You OTP is : 54606
```

BIBLIOGRAPHY

- Java: The complete Reference (11th edition), by Herbert Schildt, McGraw Hill Education
- Core Java volume 1- Fundamentals (11th edition), by Cay Horstmann, Prentice Hall