**Project Name:** “Secure Password Generator”

**Description:**

This project focuses on building a Random Password Generator using Python to generate secure passwords for various use cases. A strong password is crucial for securing online accounts and protecting sensitive information. The project uses Python’s libraries to generate random, complex passwords with customizable options such as length and character composition (uppercase letters, lowercase letters, numbers, and special characters). This project demonstrates programming fundamentals, string manipulation, and the use of libraries to ensure password security.

**Features**:

1. Customization Options:Allows users to choose the length and types of characters included in the password.

2. High Security:

Generated passwords are complex and include a mix of character types to enhance security.

3. Simple User Interface:

Command-line interaction where users input their preferences, making it easy to use.

**Technologies Used:**

Programming Language: Python 3

Development Tools: Visual Studio Code, Jupyter Notebook, or any Python IDE

Libraries: Python Standard Library, specifically random and string modules.

**Console-Based Code:**

import random

import string

# Function to generate a random password

def generate\_password(length=8, use\_uppercase=True, use\_digits=True, use\_special=True):

# Base character set

character\_set = string.ascii\_lowercase

# Add additional character types based on user selection

if use\_uppercase:

character\_set += string.ascii\_uppercase

if use\_digits:

character\_set += string.digits

if use\_special:

character\_set += string.punctuation

# Ensure at least one character from each selected type

password = [

random.choice(string.ascii\_lowercase),

random.choice(string.ascii\_uppercase) if use\_uppercase else '',

random.choice(string.digits) if use\_digits else '',

random.choice(string.punctuation) if use\_special else ''

]

random.choice(string.punctuation) if use\_special else ''

]

# Generate the remaining characters randomly

password += [random.choice(character\_set) for \_ in range(length - len(password))]

# Shuffle the password list to randomize

random.shuffle(password)

# Convert list to string and return

return ''.join(password)

# User inputs

length = int(input("Enter the desired password length (8-16): "))

use\_uppercase = input("Include uppercase letters? (y/n): ").lower() == 'y'

use\_digits = input("Include digits? (y/n): ").lower() == 'y'

use\_special = input("Include special characters? (y/n): ").lower() == 'y'

# Generate and display the password

password = generate\_password(length, use\_uppercase, use\_digits, use\_special)

print("Generated Password:", password)

OUTPUT:

