**🔐 Password Generator Project Report**

**📌 Project Title:**

**Simple & Smart Password Generator using Python**

**📄 1. Introduction**

The Password Generator is a Python-based application designed to generate strong and secure passwords. Users can select the desired password complexity and length based on their requirements. This project aims to promote better password hygiene and security awareness by creating robust, non-guessable passwords.

**⚙️ 2. Objectives**

* To design a user-friendly password generator.
* To offer multiple levels of password complexity (Simple, Moderate, and Complex).
* To estimate the strength of the password generated.
* To provide an option to save generated passwords for later use.

**💻 3. Tools and Technologies Used**

* **Programming Language:** Python 3
* **Modules Used:**
  + random – for selecting characters randomly
  + string – for accessing pre-defined character sets
  + file handling – for saving passwords to a text file

**🧠 4. Working Principle**

1. The user selects the **password length** (between 4 and 32 characters).
2. The user selects the **complexity level**:
   * **Level 1**: Letters only
   * **Level 2**: Letters and Digits
   * **Level 3**: Letters, Digits, and Symbols
3. A password is generated using random.choice() based on the selected criteria.
4. The password's **strength** is evaluated based on length and complexity:
   * Weak / Medium / Strong
5. The user is given the option to **save the password** to a local file passwords.txt.

**🧪 5. Features**

* Password length control
* Three-tier complexity selection
* Password strength evaluation
* File saving option
* Error handling for invalid inputs

**🔐 6. Sample Output (Textual Description)**

* User enters length: 10
* Selects complexity level: 3
* Output:

pgsql

CopyEdit

🔑 Generated Password: A9$dX@3rTq

🔒 Password Strength: Strong

✅ Saved to 'passwords.txt'.

**✅ 7. Advantages**

* Easy to use and extend
* Helps users follow good password practices
* Safe way to store generated passwords (optionally)

**📌 9. Conclusion**

This project successfully demonstrates the creation of a practical and user-friendly password generator. It helps promote secure digital practices by guiding users toward using strong, random passwords based on their chosen security level.