

PASSWORD GENERATOR IN PYTHON

SECURE AND RANDOM PASSWORD CREATION



INTRODUCTION:

- Purpose of the project: To generate secure passwords quickly.
- Importance of strong passwords in cybersecurity.

FEATURES:

- User input for password length.
- Mix of letters, digits, and special characters.
- Error handling for invalid input.

CODE EXPLANATION (PART 1)&(PART 11):

- `import random` and `import string``
- ``generate_password` function:`
- Combines letters, digits, punctuation.
- Uses ``random.choices()`` to pick characters.
- Taking user input and converting to integer.
- Try-except block to catch invalid input.
- Printing the generated password or error message.

SAMPLE OUTPUT:

- Show example outputs for different inputs:
- Length = 8 → Password example: `A9&k3\$zP`
- Length = 16 → Password example: `!fL7@0pS#Xvq2rTz`
- Highlight randomness and security.

BENEFITS AND USE CASES:

- Easy-to-use command line utility.
- Can be integrated into user signup systems.
- Helps users create strong passwords to improve security.

CONCLUSION:

- Recap: simple, effective password generator.
- Importance of using strong passwords.
- Invitation to extend the project (e.g., GUI, password strength checker).