Position- Python Programming Intern

S.no—Basic (silver certificate)

01- Number – Guessing Game

Ans---

```
      import random
# Set the secret number
secret_number = random.randint(1, 100)
# Initialize the number of attempts
attempts = 0
print("Welcome to the number-guessing game!")
print("I'm thinking of a number between 1 and 100.")
while True:
    # Ask the user for their guess
    user_guess = int(input("Enter your guess: "))
   # Increment the number of attempts
    attempts += 1
 # Check if the user's guess is correct
    if user_guess == secret_number:
        print(f" Congratulations! You guessed the number in {attempts} attempts.")
        break
  # Provide a hint if the user's guess is too high or too low
    elif user_guess > secret_number:
        print("Too high! Try again.")
    else:
        print("Too low! Try again.")
```

2—Currency Converter

ans=

```
class CurrencyConverter:
    def _init_(self):
        self.exchange_rates = {
            'USD': 1,  'EUR': 0.84,
            'GBP': 0.76, 'INR': 74.83,
            'AUD': 1.31,  'CAD': 1.23,
            'SGD': 1.35,  'CHF': 0.92
        }
   def convert(self, from_currency, to_currency, amount):
        if from_currency != 'USD':
            amount = amount / self.exchange_rates[from_currency]
  return amount * self.exchange_rates[to_currency]
def main():
    converter = CurrencyConverter()
    print("Available currencies: USD, EUR, GBP, INR, AUD, CAD, SGD, CHF")
   from_currency = input("From currency: ").upper()
    to_currency = input("To currency: ").upper()
    amount = float(input("Amount: "))
   result = converter.convert(from_currency, to_currency, amount)
    print(f"{amount} {from_currency} is equal to {result} {to_currency}")
if _name_ == "_main_":
    main()
```

3- password Generator

Ans=

```
import string

import secrets

def generate_password(length):

    """  Generate a strong password of a specified length.

   Args:

        length (int): The length of the password to generate.

    Returns:

        str: A strong password of the specified length.

    """   characters = string.ascii_letters + string.digits + string.punctuation

    while True:

        password = ''.join(secrets.choice(characters) for _ in range(length))

        if (any(c.islower() for c in password)

                and any(c.isupper() for c in password)

                and any(c.isdigit() for c in password)

                and any(c in string.punctuation for c in password)):

            break

    return password

# Example usage:

length = 12  # Specify the length of the password

password = generate_password(length)

print(f"Generated password: {password}")
```

Intermediate project (mandatory for gold certificate)

ALARM CLOCK

Ans=

```
import datetime
import os
import time
def set_alarm(alarm_time, sound_file):
  """ Set an alarm to go off at a specified time.
 Args:
    alarm_time (str): The time to set the alarm in HH:MM format.
    sound_file (str): The path to the sound file to play when the alarm goes off.
  """  alarm_hour, alarm_minute = map(int, alarm_time.split(':'))
  alarm_time = datetime.datetime.now()
  alarm_time = alarm_time.replace(hour=alarm_hour, minute=alarm_minute, second=0)
  print(f"Alarm set for {alarm_time.strftime('%I:%M %p')}")
 while True:
    time.sleep(1)
    now = datetime.datetime.now()
    if now >= alarm_time:
      print("Wake Up!")
      os.system(f"start {sound_file}")  # Play the sound file
      break
# Example usage:
alarm_time = "06:30"  # Set the alarm time in HH:MM format
sound_file = "alarm_sound.mp3"  # Set the path to the sound file
set_alarm(alarm_time, sound_file)
```