

```

import re

class UserInput:
    def __init__(self):
        self.name = ""
        self.password = ""
        self.email = ""
        self.n = 0

    def validate_name(self, name):
        return bool(re.match(r'^(?=.*[0-9])(?=.*[a-zA-Z0-9])([a-zA-Z0-9][^
]*)$', name))

    def validate_password(self, password):
        return bool(re.match(r'^[a-zA-Z#_@]{1,8}$', password))

    def validate_email(self, email):
        return bool(re.match(r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z
]{2,}$', email))

    def get_valid_input(self):
        while True:
            self.name = input("Enter your Name: ")
            if self.validate_name(self.name):
                break
            print("Invalid Name! It must contain exactly one number and one
special character.")

        while True:
            self.password = input("Enter your Password: ")
            if self.validate_password(self.password):
                break
            print("Invalid Password! It must not contain numbers or special
characters except (#, _, @) and must be at most 8 characters long.")

        while True:
            self.email = input("Enter your Email Address: ")
            if self.validate_email(self.email):
                break
            print("Invalid Email Address! Please enter a valid format (e.g.,
example@mail.com).")

        while True:
            try:
                self.n = int(input("How many times do you want to Print? "))
                if self.n > 0:
                    break
            print("Number must be greater than zero.")

```

```

        except ValueError:
            print("Invalid input! Please enter a valid number.")

    def display_output(self):
        print(f"{self.name} Wants to Print {self.n} Times and send Mail to {self.email}.")
        print(f"Your password {self.password} will be encrypted and will be stored.")

if __name__ == "__main__":
    user = UserInput()
    user.get_valid_input()
    user.display_output()

```

```

import re

class UserAccount:
    def __init__(self):
        self.name = ""
        self.department = ""
        self.password = ""
        self.encoded_name = ""
        self.security_questions = {
            "What is your favorite color?": "",
            "What is your birth month?": ""
        }

    def validate_password(self, password):
        return bool(re.match(r'^(?=.*[0-9])(?=.*[a-zA-Z0-9])([a-zA-Z0-9#@_]{1,8})$', password))

    def get_user_details(self):
        self.name = input("Enter your Name: ")
        self.encoded_name = self.name[:3] + "*** - Fresher"
        self.department = input("Enter your Department: ")

    def set_password(self):
        while True:
            password = input("Enter your Password: ")
            if self.validate_password(password):
                break
            print("Invalid Password! It must contain at least one number, one special character and be at most 8 characters long.")

        attempts = 3

```

```

        while attempts > 0:
            retype_password = input("Re-Type your Password: ")
            if retype_password == password:
                self.password = password
                print("Password set successfully!")
                return
            attempts -= 1
            print(f"Passwords do not match! You have {attempts} attempts
left.")

        self.forgot_password_option()

    def forgot_password_option(self):
        choice = input("Forgot Password (Y/N) or Know password?
").strip().lower()
        if choice == 'y':
            self.recover_password()
        else:
            print("You need to remember your password to proceed!")

    def set_security_questions(self):
        print("\nSet Security Questions:")
        for question in self.security_questions:
            self.security_questions[question] = input(question + " ")

    def recover_password(self):
        print("\nAnswer Security Questions to recover your password.")
        while True:
            correct_answers = 0
            for question, answer in self.security_questions.items():
                user_answer = input(question + " ")
                if user_answer == answer:
                    correct_answers += 1

            if correct_answers == 2:
                print(f"\nYour Password is: {self.password}")
                return
            print("Incorrect answers! Please try again.")

    def display_details(self):
        print(f"\nYour Encoded Name is: {self.encoded_name}")
        print(f"Your Department is: {self.department}")
        print(f"Your Password is: {self.password}")

if __name__ == "__main__":
    user = UserAccount()
    user.get_user_details()
    user.set_security_questions()

```

```
user.set_password()  
user.display_details()
```