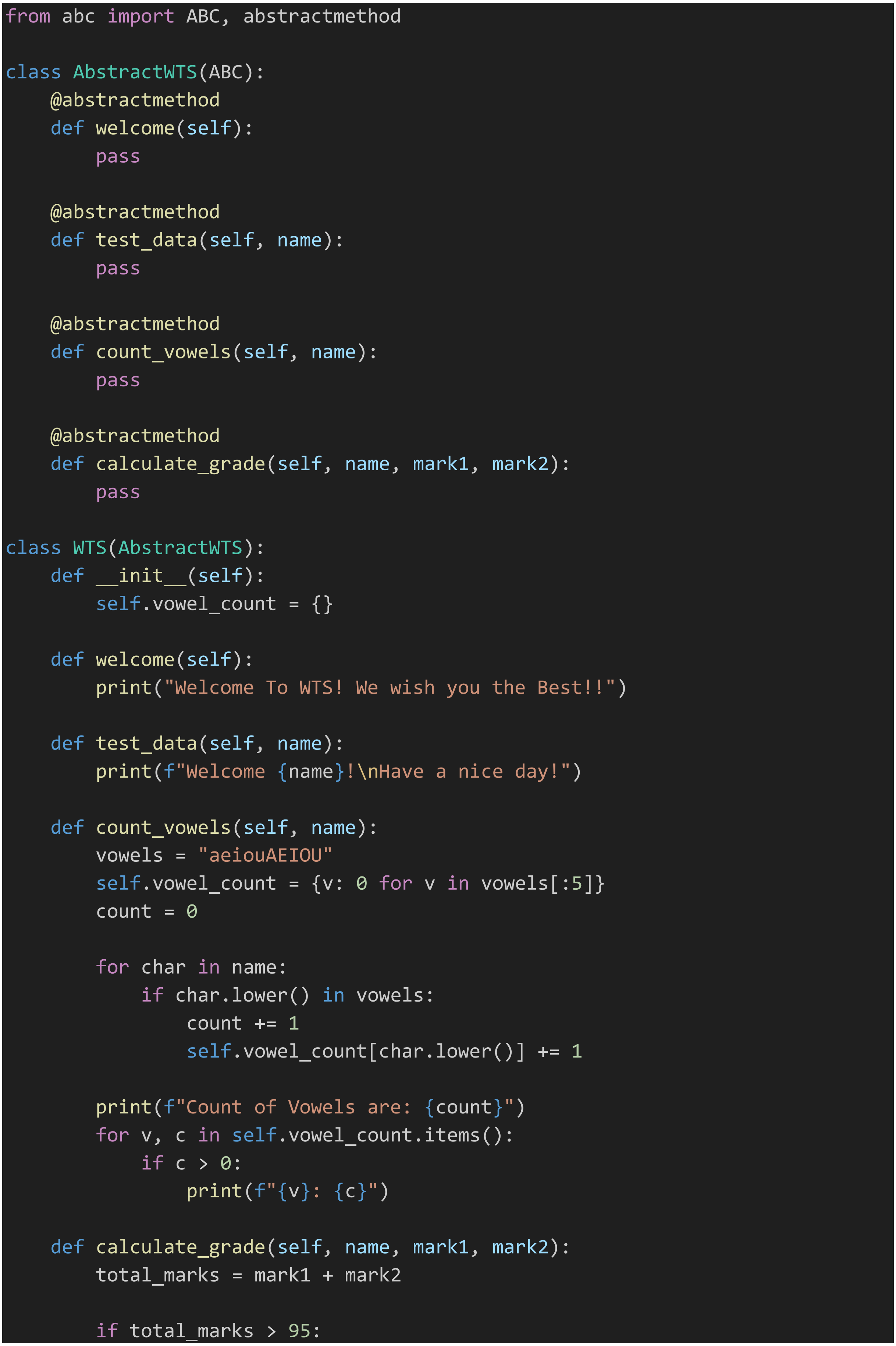

```python
from abc import ABC, abstractmethod

class AbstractWTS(ABC):
    @abstractmethod
    def welcome(self):
        pass

    @abstractmethod
    def test_data(self, name):
        pass

    @abstractmethod
    def count_vowels(self, name):
        pass

    @abstractmethod
    def calculate_grade(self, name, mark1, mark2):
        pass

class WTS(AbstractWTS):
    def __init__(self):
        self.vowel_count = {}

    def welcome(self):
        print("Welcome To WTS! We wish you the Best!!")

    def test_data(self, name):
        print(f"Welcome {name}!\nHave a nice day!")

    def count_vowels(self, name):
        vowels = "aeiouAEIOU"
        self.vowel_count = {v: 0 for v in vowels[:5]}
        count = 0

        for char in name:
            if char.lower() in vowels:
                count += 1
                self.vowel_count[char.lower()] += 1

        print(f"Count of Vowels are: {count}")
        for v, c in self.vowel_count.items():
            if c > 0:
                print(f"{v}: {c}")

    def calculate_grade(self, name, mark1, mark2):
        total_marks = mark1 + mark2

        if total_marks > 95:
            grade = "E"
        elif 80 <= total_marks <= 95:
            grade = "A+"
        elif 75 <= total_marks < 80:
            grade = "A"
        elif 60 <= total_marks < 75:
            grade = "B"
        else:
            grade = "F"

        print(f"{name}'s Grade: {grade}")

class PasswordManager:
    def __init__(self, password):
        self.password = password

    def validate_password(self):
        for char in self.password:
            if char.isdigit():
                print(f"Your Output will Break here -
{self.password[:self.password.index(char)]}")
                return
        print("Password is valid.")

    def skip_numbers(self):
        filtered_password = "".join([char for char in self.password if not
char.isdigit()])

if __name__ == "__main__":
    wts = WTS()

    wts.welcome()

    name = input("Please input a name: ").strip()
    wts.test_data(name)

    wts.count_vowels(name)

    try:
        mark1 = int(input(f"Enter the first mark for {name}: ").strip())
        mark2 = int(input(f"Enter the second mark for {name}: ").strip())
        wts.calculate_grade(name, mark1, mark2)
    except ValueError:
        print("Marks must be numerical values.")

    password = input("Enter a password to validate (no numerical values
allowed): ").strip()
    password_manager = PasswordManager(password)
    password_manager.validate_password()

    password_with_skip = input("Enter a password to skip numbers: ").strip()
    password_manager = PasswordManager(password_with_skip)
    password_manager.skip_numbers()
```