```
from random import randint, shuffle

class Player:
    def __init__(self, name, role, hp, attack):
        self.name = name
        self.role = role
        self.hp = hp
        self.attack = attack

    def __str__(self):
        return f"{self.name} ({self.role}): HP: {self.hp}, Attack: {self.attack}"

    def attackOpponent(self, opponent):
        opponent.hp -= self.attack
        print(f"{self.name} attacks {opponent.name}! {opponent.name} now has {opponent.hp}
HP.")

class Novice(Player):
    def __init__(self, name):
        super().__init__(name, "Novice", 100, 10)

class Swordsman(Player):
    def __init__(self, name):
        super().__init__(name, "Swordsman", 120, 15)

class Archer(Player):
    def __init__(self, name):
        super().__init__(name, "Archer", 100, 20)

class Magician(Player):
    def __init__(self, name):
        super().__init__(name, "Magician", 80, 25)

class Monster(Player):
    def __init__(self):
        super().__init__("Monster", "Boss", 150, 12)

class Game:
    def __init__(self):
        self.mode = None

    def startGame(self):
        print("\nWelcome to the Brawlhalla!\n")
        self.chooseMode()
```

```
        player1, player2 = self.createPlayer()
        self.playMatch(player1, player2)

    def chooseMode(self):
        mode = input("Choose game mode:\n1. Single Player\n2. Player vs Player\nEnter your
choice: ")
        self.mode = "Single Player" if mode == '1' else "Player vs Player"

    def createPlayer(self):
        if self.mode == "Single Player":
            return Novice(input("Enter your name: ")), Monster()
        else:
            return self.createPlayers(1), self.createPlayers(2)

    def createPlayers(self, playerNum):
        name = input(f"Enter Player {playerNum} name: ")
        roleMap = {'1': Swordsman, '2': Archer, '3': Magician}
        role = input("Choose role:\n1. Swordsman\n2. Archer\n3. Magician\nEnter choice: ")
        return roleMap.get(role, Novice)(name)

    def playMatch(self, player1, player2):
        print(f"\n{self.mode} Match:\n{player1}\n{player2}")
        turnOrder = [player1, player2]
        shuffle(turnOrder)

        while player1.hp > 0 and player2.hp > 0:
            for player in turnOrder:
                opponent = player1 if player == player2 else player2
                player.attackOpponent(opponent)
                if opponent.hp <= 0:
                    print (f"\n{player.name} wins the match!")
                    return

if __name__ == '__main__':
    Game().startGame()
```