Модуль main

from m1 import \*

def main():

global player, weapon\_choices, shield\_choices

print("Добро пожаловать в текстовую RPG игру!")

print("Вы проснулись в влажной пещере. Голова расскалывается.")

# Запрос имени игрока

name = input("Как вас зовут? ")

player = Player(name)

print(f"\n{name}, ваше приключение начинается!")

# Инициализация оружия и щита

weapon\_choices = [

Weapon("ножка стула", 3),

Weapon("кость для атаки", 5),

Weapon("ржавый меч", 7),

Weapon("кулак", 1)

]

shield\_choices = [

Shield("без щита", 0),

Shield("щит из стула", 1),

Shield("кожаный щит", 2)

]

while True:

print("Выберите действие:")

print("1. Продолжить исследование пещеры")

print("2. Привал (восстановить здоровье)")

print("3. Осмотреться на месте")

print("4. Выйти из игры")

choice = input("Введите номер действия: ")

if choice == "1":

explore\_cave(player)

elif choice == "2":

player.heal()

elif choice == "3":

print("\nВы решили осмотреться. В углу вы обнаружили ножку стула")

player.weapon = weapon\_choices[0]

print(player.weapon)

explore\_cave(player)

elif choice == "4":

print("\nВы решили покинуть игру. До новых встреч!")

break

else:

print("Неверный ввод. Попробуйте снова.")

def event(player, actions):

*"""*

*Функция для обработки событий с выбором действий.*

*actions: словарь, где*

*ключ - номер действия,*

*значение - кортеж (описание, тип\_действия, значение)*

*тип\_действия: 'action' - обычное действие, 'weapon' - получить оружие, 'shield' - получить щит*

*Пример {"1": "Пройти мимо", ""}*

*"""*

print("Выберите действие:")

for key, (d, \_, \_) in actions.items():

print(f"{key}: {d}")

user\_choice = input("Введите номер действия: ")

if user\_choice in actions:

d, action\_type, value = actions[user\_choice]

if action\_type == 'weapon':

player.weapon = weapon\_choices[value]

print(player.weapon)

elif action\_type == 'shield':

player.shield = shield\_choices[value]

print(player.shield)

elif action\_type == 'action':

print(value)

return user\_choice

else:

print("Неверный ввод. Попробуйте снова.")

return event(player, actions)

def explore\_cave(player):

# Сцена 1: Встреча с крысой

print("\nВы встретили крысу!")

rat = Monster("Крыса", hp=10, damage\_range=(1, 2), loot\_table=["монеты"])

if combat(player, rat):

# Сцена 2: Находка щита

print("\nВы находите остаток стула и переделываете его в щит!")

player.shield = Shield("щит из стула", 1)

print("Теперь вы защищены от 1 урона.")

# Сцена 3: Встреча со скелетом

print("\nВы сталкиваетесь со скелетом!")

skeleton = Monster("Скелет", hp=15, damage\_range=(3, 5), loot\_table=["кость для атаки", "ржавый меч", "монеты"])

event(player, { # значение - кортеж (описание, тип\_действия, значение)

"1": ("Идти дальше, кости это мерзость", "action", "Вы просто прошли рядом с поверженным противником"),

"2": ("Выбрать кость потолще", "weapon", 1)

})

combat(player, skeleton)

# Сцена 4: Встреча с гоблином

print("\nВы встретили гоблина!")

goblin = Monster("Гоблин", hp=20, damage\_range=(5, 7), loot\_table=["кожаный щит", "кинжал", "монеты"])

combat(player, goblin)

print("\nИгра окончена.")

def combat(player, enemy):

while player.is\_alive() and enemy.hp > 0:

print(f"{player}, оружие:{player.weapon.name}, щит:{player.shield.name}")

print(f"{enemy}")

# Ход игрока

print("\nВаше действие:")

print("1. Атаковать", end = ", ")

print("2. Попытаться уйти")

choice = input("Выберите действие: ")

if choice == "1":

player.attack\_monster(enemy)

elif choice == "2":

print("Вы решили попытаться уйти...")

escape\_chance = random.random()

if escape\_chance > 0.5:

print("Вы успешно сбежали!")

break

else:

print("Вам не удалось сбежать!")

else:

print("Неверный ввод. Попробуйте снова.")

if enemy.hp > 0:

# Ход монстра

enemy.attack(player)

if player.is\_alive():

if choice == "1":

print(f"\n{enemy.name} повержен!")

loot = enemy.drop\_loot()

if loot == "монеты":

coins = random.randint(10, 50)

print(f"Вы нашли {coins} монет!")

else:

if isinstance(loot, Weapon):

player.weapon = loot

print(f"Вы нашли {loot.name}!")

elif isinstance(loot, Shield):

player.shield = loot

print(f"Вы нашли {loot.name}!")

return True

return False

else:

print("\nВы проиграли... Игра окончена.")

exit()

if \_\_name\_\_ == "\_\_main\_\_":

main()

Модуль m1

import random

class Player:

def \_\_init\_\_(self, name):

self.name = name

self.max\_hp = 50 # Максимальное здоровье игрока

self.hp = self.max\_hp # Текущее здоровье игрока

self.weapon = Weapon("кулак", 1) # Изначальное оружие

self.shield = Shield("без щита", 0) # Изначально без щита

def attack\_monster(self, enemy):

print(f"{self.name} атаковал {enemy.name} и нанес {self.weapon.damage} урона")

enemy.take\_damage(self.weapon.damage)

def take\_damage(self, damage):

effective\_damage = damage - self.shield.protection

self.hp -= effective\_damage

def heal(self):

self.hp = self.max\_hp

print("Вы полностью восстановили здоровье!")

def is\_alive(self):

return self.hp > 0

def \_\_str\_\_(self):

return f"{self.name}, здоровье: {self.hp}/{self.max\_hp}"

class Monster:

def \_\_init\_\_(self, name, hp, damage\_range, loot\_table):

self.name = name

self.hp = hp

self.max\_hp = hp

self.damage\_range = damage\_range

self.loot\_table = loot\_table

def attack(self, enemy):

damage = random.randint(self.damage\_range[0],self.damage\_range[1])

print(self.name, "атаковал", enemy.name, "и нанес", damage, "урона")

enemy.take\_damage(damage)

def take\_damage(self, damage):

self.hp -= damage

def drop\_loot(self):

return random.choice(self.loot\_table)

def \_\_str\_\_(self):

return f"{self.name}, здоровье: {self.hp}/{self.max\_hp}"

class Weapon:

def \_\_init\_\_(self, name, damage):

self.name = name

self.damage = damage

def \_\_str\_\_(self):

return f"Вы подобрали {self.name}. Его урон {self.damage}"

class Shield:

def \_\_init\_\_(self, name, protection):

self.name = name

self.protection = protection

def \_\_str\_\_(self):

return f"Вы подобрали {self.name}. Его защита {self.protection}"