|  |  |
| --- | --- |
| 이름 | 신동욱 |
| 학번 | 20204101 |
| 학년 | 3 |
| 과목 | 윈도우즈프로그래밍 |

텍스트, 스크린샷, 도표, 폰트이(가) 표시된 사진

자동 생성된 설명

윈도우즈프로그래밍  
과제#7

Unit 인터페이스

public interface Unit

{

int getStrength();

string getName();

string getSkill();

int getDamage();

string getClan();

}

ProtossUnit 클래스

public abstract class ProtossUnit : Unit

{

public abstract int getStrength();

public abstract string getName();

public abstract string getSkill();

public abstract int getDamage();

int Shield;

public abstract int getShield();

private string Clan;

public string getClan()

{

return Clan;

}

public ProtossUnit(string clan)

{

Clan = clan;

}

}

DarkTemplar 클래스

public class DarkTemplar : ProtossUnit

{

string Name;

string Skill;

int Damage;

int Shield;

int Strength;

public override int getStrength()

{

return Strength;

}

public override string getName()

{

return Name;

}

public override string getSkill()

{

return Skill;

}

public override int getDamage()

{

return Damage;

}

public override int getShield()

{

return Shield;

}

public DarkTemplar() : base("프로토스") {

Name = "다크템플러";

Skill = "클로킹";

Damage = 45;

Shield = 80;

Strength = 40;

}

}

class Test

{

static void Main(string[] args)

{

ProtossUnit DT = new DarkTemplar();

int Damage = DT.getDamage();

int Strength = DT.getStrength();

int Shield = DT.getShield();

Console.WriteLine("종족: " + DT.getClan());

Console.WriteLine("이름: " + DT.getName());

Console.WriteLine("쉴드: " + Shield);

Console.WriteLine("체력: " + Strength);

Console.WriteLine("데미지: " + Damage.ToString());

Console.WriteLine("스킬: " + DT.getSkill());

Console.WriteLine();

Nexus N1 = new Nexus(40);

Nexus M1 = new Nexus(30, 30);

Console.WriteLine();

Nexus N2 = new Nexus(50);

Nexus M2 = new Nexus(300, 300);

Console.WriteLine();

}

}

텍스트, 스크린샷, 폰트, 흑백이(가) 표시된 사진

자동 생성된 설명

실행 결과

텍스트, 폰트, 화이트, 대수학이(가) 표시된 사진

자동 생성된 설명

Game 클래스

abstract class Game

{

protected int userWin, computerWin, drawMatch;

public Game()

{

userWin = computerWin = drawMatch = 0;

}

~Game()

{

recordPrint();

}

public abstract void result();

public void recordPrint()

{

Console.WriteLine(userWin + "승 " + computerWin + "패 " + drawMatch + "무승부 입니다 ");

}

}

Srp 클래스

class Srp : Game

{

int user;

int com;

Random r;

public Srp()

{

r = new Random();

}

public override void result()

{

com = r.Next(1, 4);

string result = "";

if (com == 1) result = "가위";

else if (com == 2) result = "바위";

else if (com == 3) result = "보";

Console.WriteLine("컴퓨터는 " + result + "를 냈습니다.");

if (com == user)

{

Console.WriteLine("비겼습니다.");

drawMatch++;

}

else if ((com == 1 && user == 3) || (com == 2 && user == 1) || (com == 3 && user == 2))

{

Console.WriteLine("졌습니다.");

computerWin++;

}

else

{

Console.WriteLine("이겼습니다.");

userWin++;

}

Console.WriteLine("");

}

public void play()

{

while (true)

{

Console.Write("입력하세요 [가위(1), 바위(2), 보(3), 종료(0)] : ");

int i = int.Parse(Console.ReadLine());

if (i == 0)

{

return;

}

this.user = i;

result();

}

}

}

Mjb 클래스

class Mjb : Srp

{

int usr;

int com;

int whoAttack = -1;

Random rand = new Random();

public Mjb()

{

com = rand.Next(1, 4);

}

public int check()

{

if (usr != 1 && com > usr) return -1;

else if (com == usr) return 0;

else if (usr == 1 && com == 3) return 1;

else return 1;

}

public void play()

{

Console.Write("입력하세요 [가위(1), 바위(2), 보(3), 종료(0)] : ");

int i = Convert.ToInt32(Console.ReadLine());

com = rand.Next(1, 4);

if (i == 0) return;

usr = i;

result();

}

public override void result()

{

int who;

if (com == usr)

{

Console.WriteLine("비겼습니다.");

who = 0;

}

else if ((com == 1 && usr == 3) || (com == 2 && usr == 1) || (com == 3 && usr == 2))

{

who = -1;

}

else

{

who = 1;

}

Console.WriteLine("");

string result = "";

if (com == 1) result = "가위";

else if (com == 2) result = "바위";

else if (com == 3) result = "보";

Console.WriteLine("컴퓨터는 " + result + "를 냈습니다.");

if (whoAttack == -1)

{

if (who == 0)

{

Console.WriteLine("비겼습니다.\n");

play();

}

else if (who == -1)

{

Console.WriteLine("공격자는 컴퓨터입니다.\n");

whoAttack = 0;

play();

}

else

{

Console.WriteLine("공격자는 유저입니다.\n");

whoAttack = 1;

play();

}

}

else if (whoAttack == 1)

{

if (who == 0)

{

Console.WriteLine("이겼습니다.\n");

userWin++;

}

else if (who == -1)

{

Console.WriteLine("공격자는 컴퓨터입니다.\n");

whoAttack = 0;

play();

}

else

{

Console.WriteLine("공격자는 유저입니다.\n");

play();

}

}

else

{

if (who == 0)

{

Console.WriteLine("졌습니다.");

computerWin++;

}

else if (who == -1)

{

Console.WriteLine("공격자는 유저입니다.");

whoAttack = 1;

play();

}

else

{

Console.WriteLine("공격자는 컴퓨터입니다.");

play();

}

}

}

}

테스트 클래스

class Program

{

static void Main(string[] args)

{

Console.Write("선택하세요 [가위바위보 게임(1), 묵찌빠 게임(2)] : ");

int i = Convert.ToInt32(Console.ReadLine());

if (i == 1)

{

Srp s = new Srp();

s.play();

s.recordPrint();

}

else if(i == 2) {

Mjb m = new Mjb();

m.play();

m.recordPrint();

}

}

}

특정 버전에서 소멸자가 출력되지 않는 현상을 발견하여 메인 함수를 약간 수정하여 직접 결과를 출력하게 했다

실행 결과

1. 가위바위보

텍스트, 스크린샷, 폰트, 흑백이(가) 표시된 사진

자동 생성된 설명

2. 묵찌빠

텍스트, 스크린샷, 폰트, 흑백이(가) 표시된 사진

자동 생성된 설명