

第3回 オープン授業

～条件分岐～

先週までの内容

- 変数の型
 - int - 整数 %d
 - double - 実数 %f
 - String - 文字列 %s
- Scannerの使い方
- 四則演算
 - + ... 足し算
 - - ... 引き算
 - * ... 掛け算
 - / ... 割り算の商
 - % ... 割り算の余り
 - = ... 代入演算子(イコール)

条件分岐とは

- 「〇〇のときに△△する」
 - 弾が敵に当たったとき、ダメージを与える
 - ゴールに到着したとき、クリア画面を表示する
- if 文と言う ← よく使います

条件分岐の書き方

```
if(条件式){  
    //条件に当てはまるときの処理  
}
```

- 条件式に当てはまるときに {} 内の処理が行われる

条件式

比較演算子	意味	使用例
<	より小さい	$x < 10$
>	より大きい	$x > 10$
\leq	以下	$x \leq 10$
\geq	以上	$x \geq 10$
\equiv	等しい	$x \equiv 10$
\neq	等しくない	$x \neq 10$

使用例

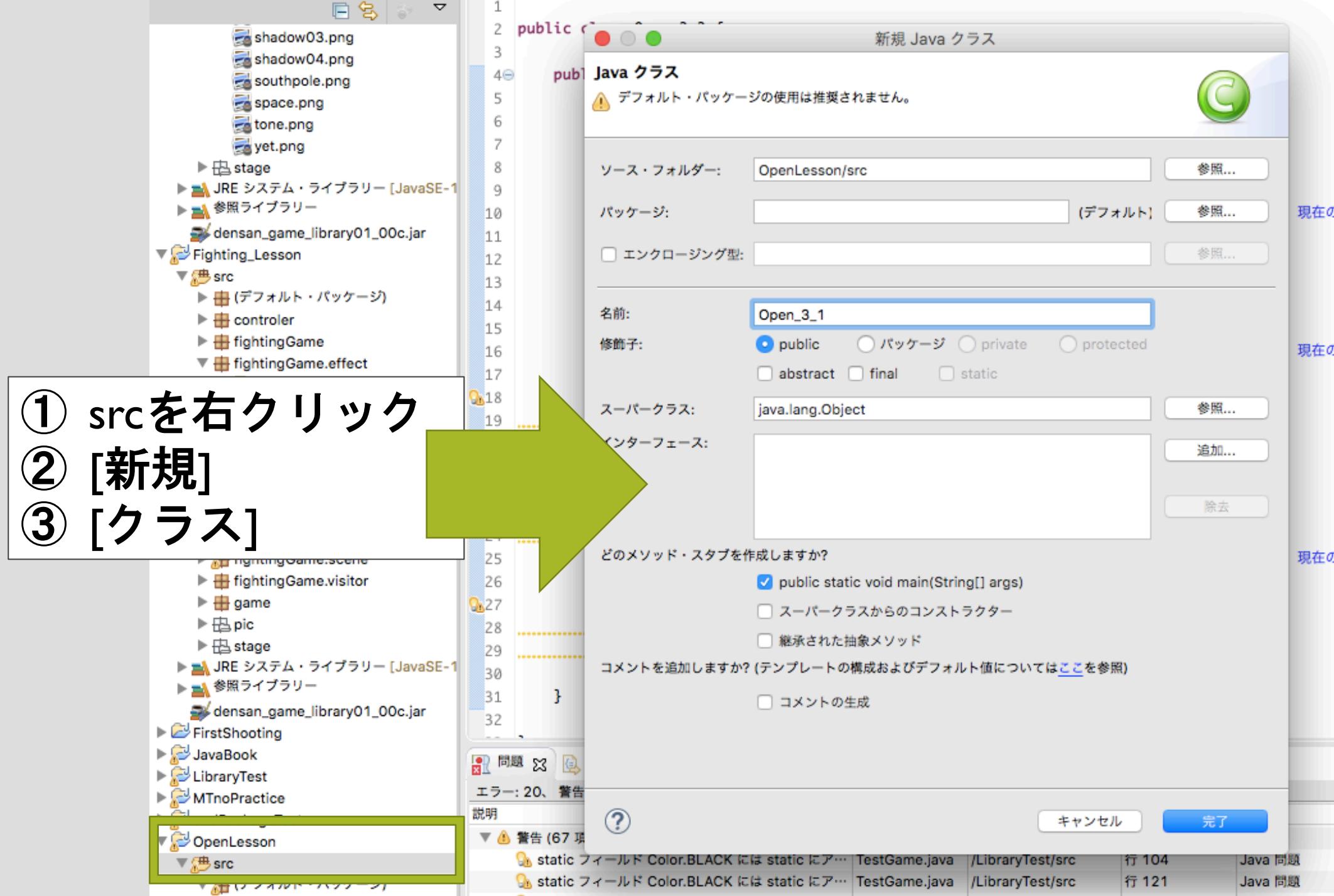
```
public static void main(String[] args) {  
    System.out.println("9の次の整数は？");  
    Scanner scanner = new Scanner(System.in);  
    int x = scanner.nextInt();  
    if(x == 10){  
        System.out.println("当たり前だよなあ");  
    }  
}
```

$x == 10$ であれば、{}内の処理が行われる

例題 1

あいさつ

- 入力された文(String型)に応じた挨拶を表示する
 - 入力する文 :greeting
- ヒント : Stringの場合、条件式で == が使えない
 - ✗ if(greeting == “おはよう”)



```
1 import java.util.Scanner;
2
3 public class Open_3_1 {
4
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         System.out.println("挨拶文を入力してください(おはよう or こんにちは or こんばんは)");
8         //入力は"おはよう","こんにちは","こんばんは"のどれか
9         String greeting = scanner.nextLine(); // nextInt()のString版的な nextLine();
10
11         if(false/* greeting がおはようの時の条件 */){
12             System.out.println("おはようございます！");
13         }
14         if(false/* greeting がこんにちはの時の条件 */){
15             System.out.println("こんにちは！");
16         }
17         if(false/* greeting がこんばんはの時の条件 */){
18             System.out.println("こんばんは！");
19         }
20         scanner.close();
21     }
22
23 }
```

実行結果

挨拶文を入力してください(おはよう or こんにちは or こんばんは)
おはよう
おはようございます！

挨拶文を入力してください(おはよう or こんにちは or こんばんは)
こんにちは
こんにちは！

挨拶文を入力してください(おはよう or こんにちは or こんばんは)
こんばんは
こんばんは！

挨拶文を入力してください(おはよう or こんにちは or こんばんは)
おやすみなさい
|

```
1 import java.util.Scanner;
2
3 public class Open_3_1_ans {
4
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         System.out.println("挨拶文を入力してください(おはよう or こんにちは or こんばんは)");
8         //入力は"おはよう","こんにちは","こんばんは"のどれか
9         String greeting = scanner.nextLine(); // nextInt()のString版的な nextLine();
10
11     if(greeting.equals("おはよう")){
12         System.out.println("おはようございます！");
13     }
14     if(greeting.equals("こんにちは")){
15         System.out.println("こんにちは！");
16     }
17     if(greeting.equals("こんばんは")){
18         System.out.println("こんばんは！");
19     }
20     scanner.close();
21 }
22
23 }
24
```

2つの条件式を使う

論理演算子	意味	使用例
<code>&&</code>	かつ (and)	$5 < x \&\& x < 10$
<code> </code>	または (or)	$x < 5 10 < x$
<code>!</code>	否定 (not)	$!(x < 10)$

使用例

```
public static void main(String[] args) {
    System.out.println("10より大きくて12より小さい整数は？");
    Scanner scanner = new Scanner(System.in);
    int x = scanner.nextInt();
    if(x > 10 && x < 12){
        System.out.println("お、 そうだな");
    }
}
```

$x > 10$ かつ $x < 12$ であれば、{}内の処理が行われる

条件にあてはまらない場合

10より大きくて12より小さい整数は？

11

お、そうだな

10より大きくて12より小さい整数は？

12

条件にあてはまらない場合は何も表示されない



else を使う

条件分岐の書き方 2

```
if(条件式1){  
    //条件式1に当てはまるときの処理  
}  
else if(条件式2){  
    //条件式1に当てはまらず、条件式2に当てはまるときの処理  
}  
.  
.  
else{  
    //全ての条件式に当てはまらないときの処理  
}
```

使用例

```
public static void main(String[] args) {  
    System.out.println("10より大きくて12より小さい整数は？");  
    Scanner scanner = new Scanner(System.in);  
    int x = scanner.nextInt();  
    if(x > 10 && x < 12){  
        System.out.println("お、そうだな");  
    }else if(x == 8){  
        System.out.println("イイっすね");  
    }else{  
        System.out.println("マズいですよ");  
    }  
}
```

10より大きくて12より小さい整数は？

11

お、そうだな

10より大きくて12より小さい整数は？

8

イイっすね

10より大きくて12より小さい整数は？

9

マズいですよ

例題 2

10未満の奇数、10以上の偶数

- 入力された整数(int型)が10未満の奇数か
10以上の偶数かを判定する
 - 入力する整数 : x
- ヒント：「2つの条件式」「条件に当てはまらない」

```
1 import java.util.Scanner;
2
3 public class Open_3_2 {
4
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7
8         System.out.print("整数を入力してください:");
9         int x = scanner.nextInt();
10
11        /*
12         * ここにif文を書く(以下を満たしてください！)
13         * ・条件分岐は3つ(条件式に当てはまらない部分も必要)
14         * ・出力(System.out.println)は実行結果に合わせる
15         * ・各if文の処理(中括弧内)は1行
16         *
17         * 奇数や偶数の条件などは検索してもらってOKです。
18         *
19         */
20
21         scanner.close();
22     }
23
24 }
```

実行結果

整数を入力してください: 1

入力された整数は10未満の奇数です

整数を入力してください: 12

入力された整数は10以上の偶数です

整数を入力してください: 6

入力された整数は条件を満たしません

```
1 import java.util.Scanner;
2
3 public class Open_3_2_ans {
4
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7
8         System.out.print("整数を入力してください:");
9         int x = scanner.nextInt();
10
11        if(x%2!=0&&x<10){
12            System.out.println("入力された整数は10未満の奇数です");
13        }
14        else if(x%2==0&&x>=10){
15            System.out.println("入力された整数は10以上の偶数です");
16        }
17        else{
18            System.out.println("入力された整数は条件を満たしません");
19        }
20
21        scanner.close();
22    }
23
24 }
```

条件分岐 まとめ

- 条件分岐の書き方
- 条件式
- 2つの条件式を使う
- 条件分岐の書き方2
- 実際に使って覚えよう！

練習問題 1

弾が敵に当たったとき、ダメージを与える

- ・ 今回は x軸(横移動) での判定のみを考える
- ・ 弾のx座標 : bulletX
- ・ 敵のx座標 : enemyX
- ・ 敵の体力 : enemyLife
- ・ 次ページ以降にあるコードを使ってください

```
public class Open_3_3 {  
  
    public static void main(String[] args) {  
        int bulletX = 0;  
        int enemyX = 10;  
        int enemyLife = 1;  
        System.out.println("現在の弾のx座標：" + bulletX + ", 現在の敵のx座標：" + enemyX + ", 現在の敵の体力：" + enemyLife);  
        System.out.println("弾を発射します(10移動);  
        /*  
         * 弾の座標を敵の座標に10近づける  
         */  
        System.out.println("現在の弾のx座標：" + bulletX + ", 現在の敵のx座標：" + enemyX + ", 現在の敵の体力：" + enemyLife);  
        if(false/* 弾が当たったときの条件 */){  
            System.out.println("弾が敵に命中しました！");  
            /*  
             * 敵の体力(enemyLife)に1ダメージ  
             */  
            bulletX = 0; // 弾の座標をリセット  
        }  
        System.out.println("現在の弾のx座標：" + bulletX + ", 現在の敵のx座標：" + enemyX + ", 現在の敵の体力：" + enemyLife);  
        if(false/* 敵の体力が0になった時の条件 */){  
            System.out.println("敵を倒しました！");  
        }  
    }  
}
```

```
1
2 public class Open_3_3 {
3
4     public static void main(String[] args) {
5
6         int bulletX = 0;
7         int enemyX = 10;
8         int enemyLife = 1;
9
10        System.out.println("現在の弾のx座標 : "+ bulletX +", 現在の敵のx座標 : "+ enemyX +", 現在の敵の体力 : "+ enemyLife);
11
12        System.out.println("弾を発射します(10移動)");
13        /*
14         * 弾の座標を敵の座標に10近づける
15         */
16        System.out.println("現在の弾のx座標 : "+ bulletX +", 現在の敵のx座標 : "+ enemyX +", 現在の敵の体力 : "+ enemyLife);
17
18        if(false/* 弾が当たったときの条件 */){
19            System.out.println("弾が敵に命中しました！");
20            /*
21             * 敵の体力(enemyLife)に1ダメージ
22             */
23            bulletX = 0; // 弾の座標をリセット
24        }
25        System.out.println("現在の弾のx座標 : "+ bulletX +", 現在の敵のx座標 : "+ enemyX +", 現在の敵の体力 : "+ enemyLife);
26
27        if(false/* 敵の体力が0になった時の条件 */){
28            System.out.println("敵を倒しました！");
29        }
30    }
31 }
32
33 }
34 |
```

実行結果

```
現在の弾のx座標 : 0, 現在の敵のx座標 : 10, 現在の敵の体力 : 1
弾を発射します(10移動)
現在の弾のx座標 : 10, 現在の敵のx座標 : 10, 現在の敵の体力 : 1
弾が敵に命中しました!
現在の弾のx座標 : 0, 現在の敵のx座標 : 10, 現在の敵の体力 : 0
敵を倒しました!
```

ここでやりたいこと

- ① 弾の座標を敵の座標に10近づける(重なる)
- ② 弾が当たった(重なった)ときの条件で if文
- ③ 敵に1ダメージ(体力が減る)
- ④ 敵の体力が0になったときの条件で if文

上記をクリアし「敵を倒しました！」を表示させる

練習問題 2

時間内にゴールに着く

- よりゲームっぽいプログラム
 - プレイヤーは「走る」か「戦う」を選ぶ
 - 走ると多く進むがダメージを受ける
 - 戦うとダメージは無いがあまり進めない
 - 目標は「時間内にゴールに着く」こと
敵を倒さなくてもいいが、走り続けると
やられてしまう

練習問題 2

時間内にゴールに着く

- 残り時間 : time
 - プレイヤーの体力 : life
 - ゴールまでの距離 : distance
 - 次の行動 : behavior
-
- 入力は3回とする(1回目,2回目,3回目ともに同じコード)
 - 結果の部分は例題2のような条件分岐

1ページ目

```
import java.util.Scanner;

public class Open_3_4 {

    public static void main(String[] args) {
        int time = 10;
        int life = 10;
        int distance = 10;
        Scanner scanner = new Scanner(System.in);
        //~~~~~ 1回目 ~~~~~
        System.out.println("残り時間：" + time + "、プレイヤーの体力：" + life + "、ゴールまでの距離：" + distance);
        System.out.println("どうする？(" + "run" + " or " + "fight" + ")");
        String behavior = ""/* scannerを使った入力 */;
        if(false/* behavior(行動)がrun(走る)のときの条件 */){
            System.out.println("プレイヤーは走った！");
            System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
            /* プレイヤーに4ダメージ */
            System.out.println("プレイヤーは4移動した！");
            /* プレイヤーはゴールまで4移動 */
        }
        else if(false/* behavior(行動)がrun(走る)のときの条件 */){
            System.out.println("プレイヤーの攻撃！敵を倒した！");
            System.out.println("プレイヤーは2移動した！");
            /* プレイヤーはゴールまで2移動 */
        }
    }
}
```

2ページ目

```
else{ //入力が無い時はただ敵に殴られる
    System.out.println("プレイヤーには何もできない!");
    System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
    /* プレイヤーに4ダメージ */
}

time -= 4; // 何をしても時間は過ぎる
System.out.println("-----");
//~~~~~ 2回目 ~~~~~
/*
 * 1回目の内容と同じ(コピペでいいです)
*/
//~~~~~ 3回目 ~~~~~
/*
 * 1回目の内容と同じ(コピペでいいです)
*/
//~~~~~ 結果 ~~~~~
/*
 * 実行結果に合わせて条件分岐を書く
*/
scanner.close();
}

}
```

```
1 import java.util.Scanner;
2
3 public class Open_3_4 {
4
5     public static void main(String[] args) {
6         int time = 10;
7         int life = 10;
8         int distance = 10;
9         Scanner scanner = new Scanner(System.in);
10
11         //~~~~~ 1回目 ~~~~~
12
13         System.out.println("残り時間：" + time + "、プレイヤーの体力：" + life + "、ゴールまでの距離：" + distance);
14         System.out.println("どうする？( \"run\" or \"fight\" )");
15         String behavior = ""/* scannerを使った入力 */;
16         if(false/* behavior(行動)がrun(走る)のときの条件 */){
17             System.out.println("プレイヤーは走った！");
18             System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
19             /* プレイヤーに4ダメージ */
20             System.out.println("プレイヤーは4移動した！");
21             /* プレイヤーはゴールまで4移動 */
22         }
23         else if(false/* behavior(行動)がrun(走る)のときの条件 */){
24             System.out.println("プレイヤーの攻撃！\n敵を倒した！");
25             System.out.println("プレイヤーは2移動した！");
26             /* プレイヤーはゴールまで2移動 */
27     }
```

```
28     else{ //入力が無い時はただ敵に殴られる
29         System.out.println("プレイヤーには何もできない！");
30         System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
31         /* プレイヤーに4ダメージ */
32     }
33     time -= 4; // 何をしても時間は過ぎる
34     System.out.println("-----");
35
36
37     //~~~~~ 2回目 ~~~~
38     /*
39      * 1回目の内容と同じ(コピペでいいです)
40      */
41
42
43     //~~~~~ 3回目 ~~~~
44     /*
45      * 1回目の内容と同じ(コピペでいいです)
46      */
47
48
49     //~~~~~ 結果 ~~~~
50     /*
51      * 実行結果に合わせて条件分岐を書く
52      */
53
54
55     scanner.close();
56 }
57
58 }
```

実行結果

残り時間 : 10、プレイヤーの体力 : 10、ゴールまでの距離 : 10
どうする? ("run" or "fight")

run

プレイヤーは走った!

敵の攻撃! プレイヤーに4ダメージ!

プレイヤーは4移動した!

残り時間 : 6、プレイヤーの体力 : 6、ゴールまでの距離 : 6
どうする? ("run" or "fight")

run

プレイヤーは走った!

敵の攻撃! プレイヤーに4ダメージ!

プレイヤーは4移動した!

残り時間 : 2、プレイヤーの体力 : 2、ゴールまでの距離 : 2
どうする? ("run" or "fight")

run

プレイヤーは走った!

敵の攻撃! プレイヤーに4ダメージ!

プレイヤーは4移動した!

プレイヤーは力尽きた...

残り時間 : 10、プレイヤーの体力 : 10、ゴールまでの距離 : 10
どうする? ("run" or "fight")

fight

プレイヤーの攻撃!

敵を倒した!

プレイヤーは2移動した!

残り時間 : 6、プレイヤーの体力 : 10、ゴールまでの距離 : 8
どうする? ("run" or "fight")

fight

プレイヤーの攻撃!

敵を倒した!

プレイヤーは2移動した!

残り時間 : 2、プレイヤーの体力 : 10、ゴールまでの距離 : 6
どうする? ("run" or "fight")

fight

プレイヤーの攻撃!

敵を倒した!

プレイヤーは2移動した!

間に合わなかったようだ...

ここでやりたいこと

- I. 「1回目」の部分を完成させる
 - ① Scanner を使った入力 (String)
 - ② 文字列の比較による条件式
 - ③ 移動やダメージの演算
2. 「2回目」、「3回目」にも同様の処理を書く
3. 結果の部分を書く
 - 時間切れ(timeが0以下)でも、入力3回で生きて到着したらクリア

上記をクリアし「**到着！ゲームクリア！！**」を表示させる

次週の予定

- ・練習問題2でやった同じ処理を繰り返す処理
(1回目, 2回目,)についてやります
- ・分からぬことがあれば前の方まで。