

## ● JKad11D 「条件式と boolean」

以下の処理を作成せよ。

- ① ふたつの整数と、そのふたつの整数を足したときの答えを入力する。
- ② 答えが合っているときは `true`、合っていないときは `false` を表示する。このとき `true` と `false` の表示は `boolean` 型の変数を使うこと。

```
boolean correct = ((ひとつめの整数 + ふたつめの整数) == 答え);    // この変数の値を表示する
```

- ③ さらに `true` のときは「正解です!」、そうでないときは「不正解です!」と表示する。

## 課題完成時の画面① (答えが合っているとき)

```
ひとつめの整数を入力してください>1
ふたつめの整数を入力してください>2
1 + 2 の答えを入力してください>3
判定は・・・true
正解です!
```

## 課題完成時の画面② (答えが合っていないとき)

```
ひとつめの整数を入力してください>4
ふたつめの整数を入力してください>5
4 + 5 の答えを入力してください>6
判定は・・・false
不正解です!
```

## ● JKad11C1 「そうだ! 動物園へ行こう! ③」※動作は JKad10D と同じ

年齢を入力すると動物園の入場料金を表示する処理を作成せよ。入場料金は以下の通り (JKad10D と同じ)。ただし論理演算子を使わず if 文のネストで対応すること。

区分	対象年齢	料金
大人料金	15 歳以上	500 円
子供料金	6 歳以上 15 歳未満	200 円
未就学児	6 歳未満	無料

## 課題完成時の画面① (18 を入力したとき)

```
そうだ! 動物園へ行こう!!
あなたの年齢を入力してください>18
大人料金: 500 円になります!
```

## 課題完成時の画面② (6 を入力したとき)

```
そうだ! 動物園へ行こう!!
あなたの年齢を入力してください>6
子供料金: 200 円になります!
```

## ● JKad11C2 「そうだ！動物園へ行こう！④」

if～else if 文を使って、JKad11C1 と同じ処理を作成せよ。

## 課題完成時の画面

(JKad11C1 と同じ)

## ● JKad11B 「そうだ！水族館へ行こう！②」 ※動作は JKad10B と同じ

年齢を入力すると水族館の入場料金を表示する処理を作成せよ。入場料金は以下の通り (JKad10B と同じ)。ただし論理演算子を使わずに if～else if 文を使うこと。

区分	対象年齢	料金
シニア料金	65 歳以上	2200 円
大人料金	15 歳以上 65 歳未満	2400 円
子供料金	6 歳以上 15 歳未満	1200 円
幼児料金	3 歳以上 6 歳未満	600 円
未就園児	3 歳未満	無料

## 課題完成時の画面① (65 を入力したとき)

そうだ！水族館へ行こう！！  
あなたの年齢を入力してください>65  
シニア料金：2200 円になります！

## 課題完成時の画面② (15 を入力したとき)

そうだ！水族館へ行こう！！  
あなたの年齢を入力してください>15  
大人料金：2400 円になります！

## 課題完成時の画面③ (6 を入力したとき)

そうだ！水族館へ行こう！！  
あなたの年齢を入力してください>6  
子供料金：1200 円になります！

## 課題完成時の画面④ (3 を入力したとき)

そうだ！水族館へ行こう！！  
あなたの年齢を入力してください>3  
幼児料金：600 円になります！

## 課題完成時の画面⑤ (2 を入力したとき)

そうだ！水族館へ行こう！！  
あなたの年齢を入力してください>2  
無料です！

各区分の境界の値 (65 と 64、15 と 14、6 と 5、3 と 2) が  
入力されたときも仕様通り動作しているか確認すること。

## ● JKad11A 「そうだ！動物園へ行こう！（団体割引②）」※動作はJKad10Aと同じ

動物園には団体割引があった！人数と年齢を入力し、入場料金を表示する処理を作成せよ。ただし論理演算子を使わずにif～else if文を使うこと。

区分	対象年齢	通常料金	30人以上	50人以上
大人料金	15歳以上	500円	450円	400円
子供料金	6歳以上15歳未満	200円	180円	160円
未就学児	6歳未満	無料		

## 課題完成時の画面①（通常の大人料金）

そうだ！動物園へ行こう！！  
何人で行きますか？>1  
あなたの年齢を入力してください>20  
大人料金：500円になります！

## 課題完成時の画面②（30人以上の大人料金）

そうだ！動物園へ行こう！！  
何人で行きますか？>30  
あなたの年齢を入力してください>15  
大人料金：450円になります！

## 課題完成時の画面③（50人以上の子供料金）

そうだ！動物園へ行こう！！  
何人で行きますか？>50  
あなたの年齢を入力してください>6  
子供料金：160円になります！

## 課題完成時の画面④（50人以上の未就学児）

そうだ！動物園へ行こう！！  
何人で行きますか？>100  
あなたの年齢を入力してください>5  
無料です！

## ● JKad11S 「三角形を判定せよ！」（JKad11Xの方が簡単かもしれない）

三角形を構成する3つの辺の長さを入力し、三角形を作ることができるかどうか判定する処理を作成せよ。さらに三角形が作れる場合は、正三角形・二等辺三角形・その他の三角形のいずれに該当するのを表示せよ。なお、辺の長さは整数で入力するものとする。また、三角形を作るための条件は以下の通り。

三角形を作るための条件： 一番長い辺の長さ < 残りの2辺の和

## 課題完成時の画面①（正三角形のとき）

三角形を作ります！  
辺1の長さを入力してください>6  
辺2の長さを入力してください>6  
辺3の長さを入力してください>6  
正三角形になります！

## 課題完成時の画面②（二等辺三角形のとき）

三角形を作ります！  
辺1の長さを入力してください>7  
辺2の長さを入力してください>7  
辺3の長さを入力してください>4  
二等辺三角形になります！

## 課題完成時の画面③（その他の三角形のとき）

三角形を作ります！  
辺1の長さを入力してください>4  
辺2の長さを入力してください>5  
辺3の長さを入力してください>6  
三角形になります！

## 課題完成時の画面④（三角形が作れないとき）

三角形を作ります！  
辺1の長さを入力してください>1  
辺2の長さを入力してください>2  
辺3の長さを入力してください>3  
三角形になりません！

## ● JKad11X 「FizzBuzz！」 (ループ処理は自分で調べること)

課題完成時の画面を参考に 1 から 40 までを表示する処理を作成せよ (FizzBuzz ゲーム)。仕様は以下の通り。

仕様 1 3 の倍数のとき「Fizz！」と表示する。

仕様 2 5 の倍数のとき「Buzz！」と表示する。

仕様 3 仕様 1 と仕様 2 の両方を満たすとき、「FizzBuzz！」と表示する。

仕様 4 上記のどれも満たさないとき、数字をそのまま表示する。

## 課題完成時の画面 (1～20)

```
1
2
Fizz!
4
Buzz!
Fizz!
7
8
Fizz!
Buzz!
11
Fizz!
13
14
FizzBuzz!
16
17
Fizz!
19
Buzz!
```

## 続き (21～40)

```
Fizz!
22
23
Fizz!
Buzz!
26
Fizz!
28
29
FizzBuzz!
31
32
Fizz!
34
Buzz!
Fizz!
37
38
Fizz!
Buzz!
```