

# 演習 1

文字の入力を受け付けます

その文字が'A'の場合は、

Aの入力を受け付けましたと表示

その文字が'B'の場合は、

Bの入力を受け付けましたと表示

その他の文字の場合は

その他の文字の入力を受け付けましたと表示する処理を作成してください

# 演習2

数値を1～100までループさせて

数値が3の倍数の場合は  
数値: Fizz

数値が5の倍数の場合は  
数値: Buzz

数値が3かつ5の倍数の場合は  
数値: Fizz Buzz

それ以外は  
数値のみ表示

と表示する処理を作成してください

# 演習3

ループの中でじゃんけんを何度もする処理を作成します。

相手は必ずグー、チョキ、パーの順で手を出し

こちらは入力を受け付けます。

入力すると: あなたの出した手: ○○、相手の出した手: ××と表示され

**相手に勝った場合:**「あなたは勝ちました」と表示されループを抜け出せる

**負けた場合は:**「あなたの負け、再チャレンジ」と表示され3回負けると「あなたは負けました」と表示されループの外に出る

**あいこの場合は:**「あいこ」と表示される

**誤った入力をした場合:** 再入力してくださいと表示します。

処理を作成します。

# 演習4

キャラクターを作成して戦わせます。

**Character**クラスを作成し、インスタンス変数として  
name(名前), hp(ヒットポイント), offence(攻撃), defense(守備)  
を持たせます。

attackメソッドを実行すると、敵インスタンスのHPが自分のoffence - 敵のdefense分減ります。  
(自分のoffense - 敵のdefenseがマイナスの場合は1)

critical\_hitメソッドを実行すると、attackメソッドの倍のダメージを与えます。

HPが0になると死んでしまい、攻撃ができなくなります

**AllCharacters**クラスを作成します。クラス変数が3個存在します  
all\_characters: 現在作成されているすべてのキャラクターの配列  
alive\_characters: 現在生きているすべてのキャラクターの配列  
dead\_characters: 現在死んでいるすべてのキャラクターの配列  
を表します。

もし同じ名前のキャラクターを登録した場合は、CharacterAlreadyExistExceptionを作成して返してください

# 演習5

継承とポリモーフィズムを用いた演習です。

Animalsクラスにspeakという抽象メソッドを定義します。

Dogクラスを作成して: speakメソッドを実行すると「わん」と表示します

Catクラスを作成して: speakメソッドを実行すると「にゃー」と表示します

Sheepクラスを作成して: speakメソッドを実行すると「めー」と表示します。

Otherクラスを作成して: speakメソッドを実行すると「そんな動物いない」と表示します。

ユーザーから入力を受け付け

1の場合はDogクラスのspeakメソッドを実行し

2の場合はCatクラスのspeakメソッドを実行し

3の場合はSheepクラスのspeakメソッドを実行し

それ以外はOtherクラスのspeakメソッドを実行します。