

1. 基礎編

1.1. 標準出力

1. Hello World!とコンソール上に表示するスクリプトを記述してください。

1.2. 変数

1. Int 型変数を宣言し、初期化してください。
2. Float 型変数を宣言し、初期化してください。
3. Double 型変数を宣言し、初期化してください。
4. Boolean 型変数を宣言し、初期化してください。
5. String 型変数を宣言し、初期化してください。
6. var と val, それぞれのキーワードによって宣言された変数の挙動の違いを説明してください。
7. 可能な限り, var と val どちらを使用する方が好ましいですか? また, その理由を説明してください。
8. 以下の宣言文では, 左辺の型指定を省略することができます。それはなぜですか?
`val num:Int = 3`
9. Non-Nullable とはどのようなことですか。
10. Nullable な変数を宣言してください。

1.3. 演算子

1. 加算, 減算, 乗算, 除算, 余りを求める演算子をそれぞれ記述してください。
2. 同値, 超える, 下回る, 以上, 以下であるか判定するための演算子をそれぞれ記述してください。
3. 演算子の挙動は何によって定義されますか?

1.4. フロー制御

1. Boolean 値に応じて処理の流れを分岐させるときに使う構文は何ですか。
2. プログラミングにおいて, 「式(expression)」と「文(statement)」の違いを説明してください。
3. C 言語などのプログラミング言語では, if は式ですか? 文ですか?
4. C 言語などのプログラミング言語で, if 文を式のように扱う構文の名前を教えてください。
5. Kotlin において, if は式ですか? 文ですか?
6. `val age:Int` が 20 以上である場合, "お酒が飲めます", そうでない場合"お酒は飲めません"という値で初期化される String 型変数を定義してください。
7. Kotlin において, 値の複数のケースに応じて処理の流れを分岐させるときに使う構文は何ですか。

1.5. 関数

1. (Int, Float) -> String のシグネチャを持つ関数を定義してください (関数の中身は記述しなくても構いません)。
2. (Float) -> Float のシグネチャを持ち, 入力された引数の値を 2 乗して返す関数を定義してください。
3. Kotlin は, 関数を第一級オブジェクトとして扱える言語です。これはどういう意味でしょうか。
4. 変数に関数を代入してください。(関数呼び出しではないことに注意)
5. デフォルト引数を持つ関数を定義してください。
6. デフォルト引数を利用するメリットを何か挙げてみてください。

1.6. クラス

1. 「自然数」を表すクラスを定義してみてください。(表現された値が Int 型で、0 未満にならなければ良いことにします)
2. 「Twitter の投稿」を表すクラスを定義してみてください。その後、いくつか値を変えて複数インスタンス化してみてください。
3. クラスに定義された値を何と言いますか。
4. クラスに定義された関数を何と言いますか。
5. アクセス修飾子を使って、プロパティカメソッドを外部からアクセスできなくしてみてください。
6. 加算演算子の演算対象となる、2 次元ベクトルを表現するクラスを定義してみてください。*
7. データ指向のクラスを定義するとき、data class を利用するメリットを説明してください。*
8. interface とは何ですか。
9. ロード中、ロード済、ロード失敗の 3 つの状態を持つクラスを、sealed class/interface を用いて定義してください。
10. Boolean や enum class ではなく、sealed class/interface を利用するメリットについて説明してください。

1.7. ラムダ

1. Int 型、Float 型を引数リストで受け取り、String 型を返すような関数の型はどのように記述されますか？
2. (Int) -> String 型ラムダを定義してください。
3. ラムダを受け取る関数を定義してください。
4. ラムダなど、関数を受け取る関数のことを何と言いますか。
5. トレーリングラムダ(trailing lambda)記法の実例を何か記述してみてください。

1.8. リスト操作

1. 1 から 100 までの整数が入った List<Int>を初期化してください。
2. List<E>クラスはあらゆる型 E に対応しますが、このようにクラスや関数で型をパラメータとして受け取る機能のことを何と言いますか？
3. リストの全ての要素を別の値に変換する、(transform: (T) -> R) -> List<R>というシグネチャを持つ Kotlin 標準の関数は何ですか？
4. ages: List<Int>を List<Boolean>に、それぞれの要素の値が 20 以上であれば True、そうでなければ False に変換してください。