



URL: <u>https://www.kkaneko.jp/pro/pi/index.html</u> (Java の基本, スライド資料とプログラム例)





アウトライン



番号	項目
	復習
17-1	クラス定義,クラス階層,継承
17-2	UMLのクラス図
17-3	UMLのクラス図に関する実演・演習
17-4	UML のクラス図からの Java プログラム 作成

各自、資料を読み返したり、課題に取り組んだりも行う この授業では、Java を用いて基礎を学び、マスターする





Java などのプログラミング言語の体 験,演習ができるオンラインサービス

GDB online

http://www.pythontutor.com/

オンラインなので、「秘密にしたいプログラム」を 扱うには十分な注意が必要 3

GDB online で Java を動かす手順



① ウェブブラウザを起動する

② 次の URL を開く

https://www.onlinegdb.com

Q https://www.onlinegdb.com

③ 「Language」のところで、「Java」を選ぶ



SPONSOR Slack — Bring your team together with Slack, the collaboration hub for work. • Debug C Share H Save 1 Run Stop { } Beautify Language -- select -- \sim source code - select -С C++ Welcome to GDB Online. GDB online is an online compiler and debugger tool for C, C++, Python C_{++} 14 C#, VB, Perl, Swift, Prolog, Javascript, Pascal, HTML, CSS, JS 5 C L L 17 Code, Compile, Run and Debug online from anywhere in world. 6 Java Python 5 #include <stdio.h> PHP 11 int main() C# 12 - { VB printf("Hello World"); 13 14 HTML, JS, CSS 15 return 0; Ruby } 17 Perl Pascal R Fortran Haskell Assembly(GCC) **Objective C SQLite**

④ ソースコードを入れる



		^	► Run	🛈 Debug	Stop	🔁 Share	💾 Save	{ } Beautify	±	Language	a Java	\sim
7 OnlineGDB beta	online compiler and debugger for c/c++ 1 public class Main											
c/c++												
code. compile. run. debug. share.		£	public int int	static v x = 100 v = 200	oid main ; :	(String[] args)	{				
IDE				tem.out.	, printf("	'%d∖n" , ×	(+ y);					
My Projects		ß	}									
Classroom new												
Learn Programming	~	<u>,</u>					input					
Programming Questions	Com	- mano	lline									
Sign Up												
Login	Standard Input: Interactive Console											
<												

5 実行.実行結果を確認 「Run」をクリック.





17-1. クラス定義, クラス階層, 継承

7





・基本データ

データの種類	基本データ型	サイズ
整数	byte	8 bit
	short	16 bit
	int	32 bit
	long	64 bit
浮動小数	float	32 bit
	double	64 bit
文字	char	16 bit
true/false	boolean	

・基本データの配列

・**クラス**に属するオブジェクト: String クラスなど多種





クラス定義の中には、属性の定義(属性名とデータ 型)、コンストラクタの定義、その他メソッドの定 義を含める.

1	<pre>class C {</pre>				
2	<pre>int qty;</pre>				
3	<pre>double weight;</pre>				
4	String name;				
5	<pre>public C(int qty, double weight, String name) {</pre>				
6	<pre>this.qty = qty;</pre>				
7	<pre>this.weight = weight;</pre>				
8	<pre>this.name = name;</pre>				
9	}				
10	}				

オブジェクトの生成を行う**メソッド**のことを **コンストラクタ**という





クラス階層とは, <u>複数のクラスが親子</u> <u>関係をなす</u>こと







・継承とは、スーパークラスの属性とメソッドをサブ クラスが受け継ぐこと



クラスの親子関係

```
class Ball {
 double x;
 double y;
 String color;
 public Ball(double x, double y,
String color) {
  this.\mathbf{x} = \mathbf{x};
  this.y = y;
  this.color = color;
 public void move(double xx, double
yy) {
  this.x = this.x + xx;
  this.y = this.y + yy;
 public void reset() {
  this.\mathbf{x} = \mathbf{x};
  this.y = y;
```

クラス名 Ball

属性 x, y, color メソッド move, reset



class Circle extends Ball {

double r:

double r) {



クラス名 Circle 属性 x, y, color, r メソッド move, reset

クラス Circle は, 親クラスで あるクラス Ball の属性とメ **ソッドを継承**する.



17-2. UML のクラス図





- ・プログラム(ソースコード)に関する**設計図の書き方の1つ**
- ・クラスと相互関係, クラスの属性, クラスの操作を書く



15

UML



 • UML は, Unified Modeling Language のこと
 • モデリング(構造や, 振る舞いや, インタラク ション)の書さ方(表現)の方法を統一すること





• **クラスの相互関係**には何があるのでしょうか?

親クラス・子クラス

関連: ある**クラス**が,別の クラスの<u>振る舞いや属性を呼び出す</u>

親クラス・子クラス









「人」についてのデータ処理をするとき, ^の 「住所」の属性や振る舞いを呼び出す場合がある

- ・「住んでいる」:呼び出す理由
- ・「0.1」や「1」: 関連の多重度







指定方法	意味
01	0または1
1	必ず1
0*	0以上
1*	1以上
n	必ずn(ただしn > 1)
0n	0以上n以下(ただしn >
	1)
1n	1以上n以下(ただしn >
	1)



・1人の「人」には,住所は1つ
 ・住所は

誰もいなくなっても(引っ越しなど) 記録は残す.同居で複数人いても記録は まとめない クラス



	۸ 🗉		□ 住所	クラス名
	氏名		番地	
	携帯電話番号		▶ 市	
	電子メールアドレス	(住んでいる	県	▲ 馬性
	駐車場の申し込み		郵便番号	
	A		間違いがないか確認	操作
□ 学生		教員		・クラスがどういう
学生番号	給与			役割を担つか
平均点	教員	番号		・メソッド名とタイプ
卒業論文研究				を「move: void」 のように書くことも

UMLの特徴



- •UML では,**書き方(表現)の方法を統一**するので, 「人によって書き方の方法が**違う**」,「使用する プログラミング言語によって書き方の方法が**違** う」ということが**ない**
- クラス、パッケージ、オブジェクト、アクティビ ティ、ユースケース、ステートチャート、シーケ ンス、コラボレーションの書き方(表現)の書き 方が統一される

※この授業では、UMLのクラス図の演習を実施



17-3. UML のクラス図に関する実 演・演習





資料: 25~26

【トピックス】 ・UMLのクラス図



オンラインのサイト www.draw.io



- <u>https://www.draw.io/</u>
- チャート、設計図等を作成できるオンラインのサ
 イト
- ・さまざまな**シェイプ (Shape)** を**組み合わせ**て、 チャート、設計図を作成
- ・画像(PNG, JPEG など)、PDFファイル形式、
 XML形式でのエクスポート機能もある





•Webブラウザで開く

https://www.draw.io

・保存先が設定できる.保存する必要がないので
 「Decide latet」を選ぶ

Save diagrams to:



次のような画面が出ることがある(初めて使用する)

• 新規作成したいので「Create New Diagram」を選ぶ



• 種類は「Brank Diagram」を選び、「Create」をクリック







シェイプの種類



全選択: CTRL + A (同時押し) ※ CTRL は「コントロールキー」



削除: メニューの「Edit」の「Delete」





アンドゥ:メニューの「Edit」の「Undo」





UML図のためのシェイプは「UML」の下 展開 JML ボタン interface» Name + item: attribute Title chrystations Component Module package **_)**_ Entity Object Control Object Q Use Case R スクロール



クラス



親クラス・子クラス





33

 ∇

Spacing

one

-Þ

___>

UML図のためのシェイプは「**UML**」の下



Freehand ...

関連

「Association 1」が使いやすい





「+」,「Text」と操作し 「住んでいる」は, キーボードで入れる 34



右クリックメニューで

□Delete





右クリックメニューで

□ Duplicate







次のクラス図を作成しなさい





17-4. UML のクラス図からの Java プログラム作成



演習

資料:39~43

【トピックス】

- ・UMLのクラス図
- ・プログラム作成



次のクラス図の Java プログラムを演習 Person -Address name: String street: String 0..1 email: Sring city: String register(Address): void printout(): void







② 続けて,次のように入れる



```
12 - class Person {
        String name;
13
14
        String email;
15
        Address address;
        public Person(String name, String email) {
16 -
            this.name = name;
17
18
            this.email = email;
19
        public void register(Address a) {
20 -
21
            this.address = a;
22
23
```







④ 続けて,次のように入れ,実行し,実行結果を確認する 🏒







・Java プログラミング入門

GDB online を使用

https://www.kkaneko.jp/pro/ji/index.html

・Java の基本

Java Tutor, GDB online を使用

https://www.kkaneko.jp/pro/pi/index.html

・Java プログラム例

https://www.kkaneko.jp/pro/java/index.html