

Week 6

Monday, October 2, 2023 11:48 AM



NOTE: 50% of grade on assignment will be taken off if a return statement, break, or continue is used within a loop.

so NEVER DO IT



Animal

Superclass
base
general

Subclass is an extension of a base

Dog is an animal

Dog

Subclass
derives
specialized

The base class will always be created first, then the subclass.

Aggregation, composition, association, and inheritance are syntax based patterns.

Singleton are design patterns that focus specifically on the function of the classes.

Java will only give an implicit default constructor if no other constructors are made. If an n-argument constructor is made, then that default construct will not be given, and there will be an error.



Day2

Wednesday, October 4, 2023
12:01 PM

Abstraction: Masking implementation details from the user.
"Extract implementation away from the user."

Static type and dynamic type

Every variable has a static type. It is its data type.

- Variable's declaration
- Compiler uses the static type to index check.

The dynamic type of a variable is the data type of the object it refers to.

- Data type of the object the variable refers to.
- Only makes sense when talking about objects.

4 Pillars
Encapsulation
abstraction
Inheritance
Polymorphism

```

Animal Animal a = new Dog();
Dog a.setBreed("");
Cat a.setGuard("");

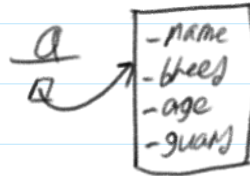
```

Animal a = new Dog();
 static type → dynamic type →

```

Animal a = new Dog();
if (a instanceof Dog) {
  ((Dog) a).setGuard(true);
}

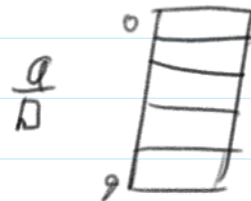
```



```

Animal[] a = new Animal[10];
a[0] = new Dog();
a[1] = new Cat();

```



Overloading: Two methods with the same name and same scope but different signatures
 (# of parameters or parameter data types)

- Signatures for java: Parameters, and return type.

Overriding: Two methods with the same signatures within an inheritance structure.
 - They need to be virtual methods. Default in java is virtual.

Hiding: Needs a distinction between nonvirtual and virtual methods. (WORRY ABOUT LATER)

Class invariant: When a decision is made that effects the actions taken by all class methods.
 - Example: A door can't be locked unless it's already closed.

HW 3 Due next Friday

Think about data types for balance

Don't think about main till the 4 classes are created.

Remember to make the classes complete! Add all possible methods.

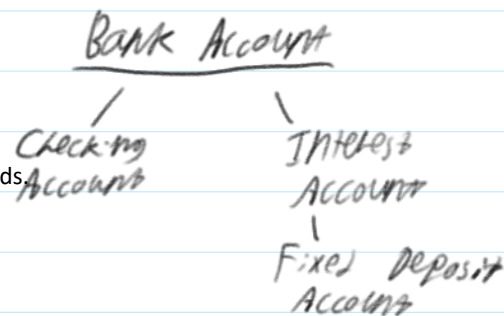
Override the toString and equals methods.

Create a container object to access the 3 derived classes.

HW 3 will not have any access classes.

What are we expected to compare for equals?

- For equals, we need to make sure that every member variable is equals. Not just balance.



For FixedDeposit, other method checks if requirementsMet is true or false, and requirementsMet is a simple setter/getter for simplicity.

QUESTIONS

Are there any requirements for the beginning check number? (As in, does it have to start at 0, or can we start it wherever the user wishes)

- ANSWER: Whatever the user wishes.

On that note, should the user be able to see the last check number?	
Should it be allowed to be mutated / accessed publicly?	

ANSWER: toString should print all variables.	
--	--

Do we need to perform any sort of input validation for the account number leading to an account of the correct class? Or perhaps input validation to check if it leads to an account at all? (using instanceof and findAccount)

- For example, when the program ask for the checking account number to deposit to, should the program make sure that the account number leads to a checking account and not an interest account?
- ANSWER: Don't worry about input validation!

The menu implementation instructions don't say anything about having an option to add interest. Does this mean interest is added implicitly when depositing or withdrawing? When should interest be added to the account's balance?

- Nope don't bother.

Why is it that we are required to type-cast for all the methods of a sub-class when using polymorphism for all the methods except toString()?

- Static binding vs Dynamic binding?

Are we allowed to add an extra member variable to give better clarification/syntax when printing?

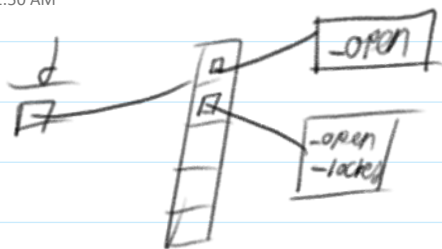
- Nope! Remove it or lose points.

The equals() method is not able to be used in main. Is this another example of the OOD rule of completion?

- Yep!

Day3

Friday, October 6, 2023
11:50 AM



d.get(i).toString()
Static type: Door

Static Binding: Method that executes is bound at compile time

Dynamic Binding: Method that executes is bound at runtime

Binding: The machine code in memory (in the function) is bound to the call of that function. This is done either statically via compile time, or dynamically during runtime. !

Compiler syntax check (Compiler)

Conversion to machine code and execution (runtime)

Dynamic binding requires an overridden method.

The compiler doesn't read code, only checks syntax.