Wednesday, November 17, 2021 10:42 AM

showBill(Product[] arr)
{
Arr[i] ========> product

Sysout (arr[i]) ====> using toString of Toy is much easier as compared to using toy getters

instanceof toy
((Toy)Arr[i]). Toyspecific methods

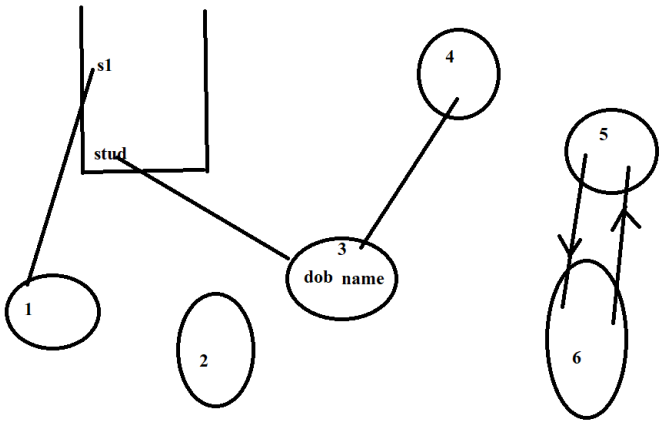
}

Garbage Collection

In C language ---------- It is the **responsibility of the programmer to release allocated memory of heap**
malloc = memory is allocated on the heap
free = memory is released from the heap

If programmer is given responsibility to release memory then - UNCERTAINTY
Following problems --------
Memory may not be efficiently managed
Memory leak
Dangling/Phantom pointers

Java proposed that responsibility of releasing the heap memory is SHIFTED from programmer to JVM!!!
JVM has a module GC (garbage collector) that is responsible for releasing heap memory !!!
A thread called as GC thread runs in the background PERIODICALLY
And frees all the objects that are not referenced directly/indirectly from the PROGRAM STACK (these are UNREACHABLE)



1 = 1 directly referenced from program stack
2 = 2 is unreachable
3 = 3 is directly referenced from program stack
4 = 4 is indirectly reachable from program stack
5, 6 = Unreachable

MARK AND SWEEP GC----------
2,5, and 6 are ELIGIBLE for GC !!! These are MARKED for GC in the first run !!
In the next run the MARKED objects are SWEEPED

Object class
public void finalize() { }

Just before SWEEPING The object the **GC calls** the **finalize**() method!!!!
In this method u can OVERRIDE and add code that is to be executed before object ends ---
Saving property values to files or database, or setting log files, setting counters!!!

JVM = will run the gc mostly when the heap is getting full !!!!
= calling finalize() may or may not happen as gc execution may or may not happen

---

Programmer may **request JVM** to run the gc by using an API ---- System.gc() !!!

How to make object unreachable
1. Define objects in small scope --- so object will go out of scope , then unreachable - eligible for gc
2. If objects are of larger scope then make them null whenever use is over
3. reasssign the reference with new object ---older object is unreachable -eligible for gc

HW --- PLAY with the GC code done in the class
1. Use array --- observe that GC is not able to free a single object , so program crashes
2. Use local scope reference --- observe that when heap starts filling jvm runs gc, gc frees unreachable objects and calls finalize
3. Use few objects in loop and see that gc is never called by JVM
4. Use System.gc() to explicitly call gc

Exception ------ PROBLEM
What happens when exception occurs --- program crashes
We want to PREVENT program crash by exception handling !!!

Two Level Problems --
1. Compile time problems = SYNTAX ERROR ===== .class is not created if syntax error occrs
2. Run time problems = problems that occur when program runs ---
    1. JVM Errors = StackOverFlowError , OutOfMemoryError
        These cannot be Handled !!! --- If they OCCUR program CRASH !
    2. Java Exceptions
        Logical problems
        or resource not available at run time ,
        or resource not available in desired format at runt time
        By default Program CRASHES -----But these can be handled at run time if logic is provided
        SO program can be PREVENTED !!

What do you mean by Program CRASHING ??? Program Terminates Abruptly
OPPOSITE
Program terminates smoothly /successfully program runs up to the last possible line in the flow !!!

Exception Handling ----------
try , catch, finally , throw , throws } Keywords

Exception in Java is a CLASS !!!
Java. Lang .Exception
This is SUPER class of all Exception

Object
Throwable
Error Exception
MANY MANY Exceptions

_____

Exception RAISED / THROW = object of the Exception or its SubClass is Created
Exception CAUGHT / CATCH === if it is caught then it is handled !!!

_____

One Try can have many catches !!!!
But **base class catch must be in the end**
OTHER WISE it will match all throws , so others are not used - hence compiler gives error

_____

throw keyword can be used in our program if we want to raise exception EXPLICITLY on some condition

throw objectOfException

2 categories of Exceptions ---- depending on if **compiler** is checking the code

1. CHECKED EXCEPTIONS ---
   If a method throws exception then CALLER MUST Catch it or DECLARE THROWS !!!
   All subclasses of Exception class except RunTimeException classes
2. UNCHECKED EXCEPTIONS
   If compiler is not having any compulsions
   All Subclasses of RuntimeException

HW --------

1. Try the CrashingProgram and All the Handling programs by typing on your machine
2. Write a class TestCommandLine Class in study.exceptions package
   1. Show the first string passed in command line in uppercase
   2. Show the square of number passed as second command line argument

   Observe program crash when user does not give any command line input
   Observer program crash if user does not give a number in second command line argument

   Handle the exceptions and Prevent crashing
   You may write different try catch for first string processing and second number processing

_____

Custom Exception ----------
WrongDayException ------ when anyone enters day of month not in 1 to 31
WrongMonthException

HW3 ---------

Write the MyDate class in study.exception.MyDate2
Property --- day, month,year
2 constructors, getter setter toString

In the parameter constructor
If day is not in range throw new WrongDayException
If month is not in range throw new WrongMonthException

Same for setDay , setMonth methods

Write a user class in same package
Main
Create date object with correct values , incorrect values
Set day and month with correct and incorrect values
Observe exceptions
Handle exceptions

_____

}

X = new Y (new Z())

X = null (y and z are unreachable)

Exception Handling