200BenchersOnly: language instructions

Basically, everything is gym terms and it is similar to Java in that it has a form of semicolon at the end of expressions and curly braces for scoping

1. { = leftWeightClip
2. } = rightWeightClip
3. ; = pump
4. Types:
    a. lightWeight: ints under 200
    b. weight: ints over 200 but less than 1000
    c. samSulek: ints over 1000
    d. ryanBullard: ints < 0
    e. True: getItUp
    f. False: failedRep
    g. String: cables
    h. long: pr
    i. float: smallPlate - only 2.5 value allowed
5. Variables
    a. Can only be muscle groups (chest or pecs, shoulder or delts, back or lats, bicep, tricep, abs, quads, hamstring, glutes, calf, forearm, maybe allow upperChest and lowerChest, and upperBack and lowerBack, obliques)
6. Variable Assignment (=): loadBar

7. Integer Operations:
    a. addition: creatine
    b. subtraction: restDay
    c. multiplication: steroids
    d. division: vegan
    e. modulus: muscleMass
8. Integer Operations: increment (++) - <var> superset
9. Boolean Operators:
    a. and: crushed
    b. not: spotter
    c. or: settle
10. comparison operators:
    a. greater than: biggerThan
    b. less than: smallerThan
    c. equal to: sameSize
11. Conditionals:
    a. If: canYouLift (<some boolean value or expression>) leftWeightClip

        <stuff here>
    b. else: rightWeightClip yourAFailureSo leftWeightClip

        <stuff here>

    rightWeightClip
12. Loops:
    a. set <var>, <int val> to <int val> leftWeightClip
        <stuff in loop>

rightWeightClip

13. Functions: workout <type> <var>(<params>) leftWeightClip

<stuff in here>

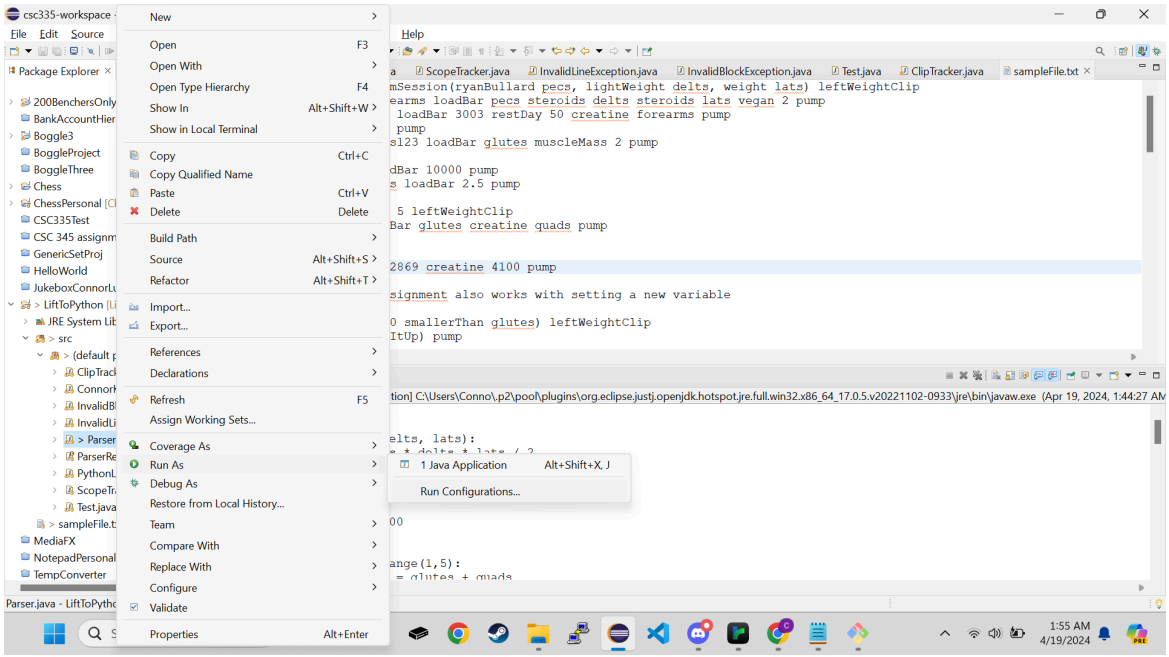
rightWeightClip

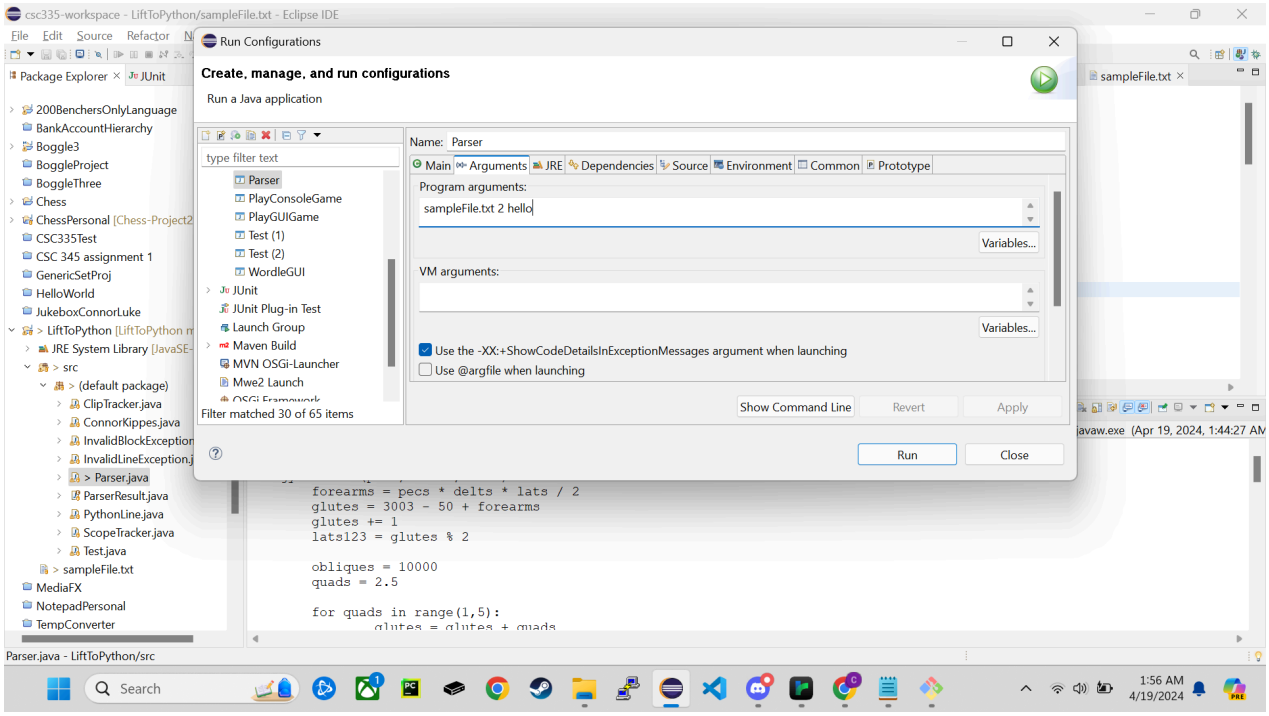
14. print(): showoff()

HOW TO RUN:

Write the programs in the sample.txt file and then right click parser class, go into run as-> run configurations -> arguments

and then add the arguments starting with sample.txt and followed by any command line arguments you want separated by single spaces. Once you're done click run.





If you want to go line by line and run in debug mode, you can remove all the arguments, run it, and just enter in a line each time and it will give you feedback, ending with entering “leave gym” to get the python code converted back for you.

SAMPLE PROGRAM:

Here is a sample program we wrote with its conversion into python:

```
workout samSulek gymSession(ryanBullard pecs, lightWeight delts, weight lats)
leftWeightClip
        ryanBullard forearms loadBar pecs steroids delts steroids lats vegan 2 pump
        samSulek glutes loadBar 3003 restDay 50 creatine forearms pump
        glutes superSet pump
        lightWeight lats123 loadBar glutes muscleMass 2 pump

        pr obliques loadBar 10000 pump
        smallPlate quads loadBar 2.5 pump

        set quads, 1 to 5 leftWeightClip
                glutes loadBar glutes creatine quads pump
        rightWeightClip

        glutes loadBar 2869 creatine 4100 pump

        sayToGymBro reasignment also works with setting a new variable

        canYouLift (2850 smallerThan glutes) leftWeightClip
                showoff(gotItUp) pump
        rightWeightClip

        canYouLift (glutes biggerThan 200) leftWeightClip
                showoff("Good work") pump
        rightWeightClip yourAFailureSo leftWeightClip
                showoff("horrible lifting") pump
        rightWeightClip

        gains glutes pump
rightWeightClip
```

```
samSulek hamstrings loadBar gymSession(-1,1,200) pump
sayToGymBro line 1: forearms =-1 * 1 * 200 / 2 = -100
sayToGymBro line 2: glutes = 3003 - 50 + (-100) = 2853
sayToGymBro line 3: glutes++ = 2854
sayToGymBro line 4: pecs123 = 2854 % 2 = 0
sayToGymBro loop: 2851 + 1 + 2 + 3 + 4 + 5= 2869
sayToGymBro afterloop: 2869 + 4100 = 6969

sayToGymBro showoff(sameSulek) pump
sayToGymBro this should print good work and return 6969


workout cables workoutPlan() leftWeightClip
        cables delts loadBar "repeat " pump
        delts loadBar delts steroids 3 pump

        cables abs loadBar "100 situps, " pump
        cables pecs loadBar abs creatine "100 pushups, " pump
        cables quads loadBar pecs creatine "100 squats, " pump
        cables calves loadBar quads creatine "and a 10 km run a day" pump

        cables biceps loadBar "saitama" pump

        canYouLift (calves sameSize "100 situps, 100 pushups, 100 squats, and a 10 km
run a day") leftWeightClip
                gains "saitama" pump
        rightWeightClip

        gains "not " creatine biceps pump
rightWeightClip

cables biceps2 loadBar workoutPlan() pump

showoff(biceps2) pump
sayToGymBro this should print saitama

workout tryBench isA200Bencher(tryBench pecs, tryBench triceps) leftWeightClip
        tryBench biceps loadBar failed pump
        canYouLift (pecs crushed triceps settle biceps) leftWeightClip
```

```
        gains gotItUp pump
    rightWeightClip

    gains spotter gotItUp pump
rightWeightClip

isA200Bencher(gotItUp,gotItUp) pump

showoff(isA200Bencher(gotItUp,gotItUp)) pump
sayToGymBro should be true

showoff(isA200Bencher(failed,gotItUp)) pump
sayToGymBro should be false
```

PYTHON CODE:

```
preworkout1 = 2
preworkout2 = "hello"
def gymSession(pecs, delts, lats):
    forearms = pecs * delts * lats / 2
    glutes = 3003 - 50 + forearms
    glutes += 1
    lats123 = glutes % 2

    obliques = 10000
    quads = 2.5

    for quads in range(1,5):
        glutes = glutes + quads

    glutes = 2869 + 4100

    # reasignment also works with setting a new variable

    if 2850 < glutes:
        print(True)

    if glutes > 200:
```

```
            print("Good work")
        else:
            print("horrible lifting")

        return glutes

hamstrings = gymSession(-1,1,200)
# line 1: forearms =-1 * 1 * 200 / 2 = -100
# line 2: glutes = 3003 - 50 + (-100) = 2853
# line 3: glutes++ = 2854
# line 4: pecs123 = 2854 % 2 = 0
# loop: 2851 + 1 + 2 + 3 + 4 + 5= 2869
# afterloop: 2869 + 4100 = 6969

# showoff(sameSulek) pump
# this should print good work and return 6969


def workoutPlan():
        delts = "repeat "
        delts = delts * 3

        abs = "100 situps, "
        pecs = abs + "100 pushups, "
        quads = pecs + "100 squats, "
        calves = quads + "and a 10 km run a day"

        biceps = "saitama"

        if calves == "100 situps, 100 pushups, 100 squats, and a 10 km run a day":
            return "saitama"

        return "not " + biceps

biceps2 = workoutPlan()

print(biceps2)
# this should print saitama

def isA200Bencher(pecs, triceps):
```

```
        biceps = False
        if pecs and triceps or biceps:
                return True

        return not True

isA200Bencher(True,True)

print(isA200Bencher(True,True))
# should be true

print(isA200Bencher(False,True))
# should be false
```