CSE 291 I: Usability of Programming Languages ("Programmers Are People Too")

Michael Coblenz

Today

- Discuss "Language Wars" paper
- A brief tour of qualitative methods
- Intro to running studies

Language Wars

- Overall impressions
- What constitutes *evidence?*
    - "Further, Boo allows the programmer to turn off the static type system (so-called Duck Typing), a decision not supported by the literature on type systems."
- How many languages do we need?
- Which RQs should we focus on?

Research Methods

Or: How We Can Obtain *Evidence*

QUALITATIVE STUDIES

- Want to understand something we don't understand yet.
  - What problems do factory workers have?
  - What is it like to write code for Indy 500 cars?
  - What usability problems do people have when they use my "awesome" system?

STAGES

- I don't know what I'm doing.
  - What problems are there to solve?
  - What hypotheses are worth testing?
- I have a tool. Let's make it better.
- I have a tool. Can people use it?
- I have a tool. Let's try to show that it IS better.

Qualitative studies

A TOUR OF QUALITATIVE METHODS

- Data sources
  - Interviews and focus groups
  - Usability studies
  - Surveys
  - Contextual inquiry
  - Corpus studies

- Analytic approaches
  - Thematic analysis
  - Grounded theory

INTERVIEWS AND FOCUS GROUPS

- Method: make a list of questions. Ask them 1-1 or to a group.
- Useful when you want to learn from experts
- Results depend on interview skill and quality of participants

USABILITY STUDIES

- Method: ask participants to do tasks with a system. Observe what problems they have.
- RQ: "What challenges do users have when they do X?"
- Great for iterating on designs
- Depends on availability of suitable users and tasks

SURVEYS

- Useful for gathering data from many people
- Not great for depth

CONTEXTUAL INQUIRY

- Watch someone doing a task
- Depends on finding an expert

CORPUS STUDIES

- RQ: "How often does X occur in the wild?"
    - or: "Does X ever occur in the wild?"
- e.g., X = null pointer dereference bugs
- e.g., X = harassment of open-source contributors
- Requires an X detector (maybe manual analysis) and a corpus

ANALYSIS

- Many qualitative studies produce textual data
  - Interview transcripts
  - Bug reports
  - Code snippets
  - Images
- Can we do better than "I read it and it seems to me…"?

OPEN-CODE THE DATA

- Meaning: categorize each element
- Manual process
- Can parallelize (have multiple coders)
    - Then have to worry about consistency
- Now you have categories!

THEMATIC ANALYSIS

- (danger: this summary is incomplete)
- In brief: repeatedly group codes until you have a hierarchy
- Top-level groups are "themes"

GROUNDED THEORY

- (danger: this summary is incomplete)
- Goal: study codes and data *deeply* until a theory emerges
  - The theory should be "grounded" in the data

Key Takeaway: Methods Answer Specific Questions

Running Studies

STUDY DESIGN OVERVIEW

- Running any kind of studies requires:
  - Ethics approval
  - Recruiting
  - Training
  - Task design
  - Data collection/analysis

ETHICS REVIEW

- For research: need to submit proposal to Institutional Review Board (IRB)
- For class: no need to get IRB approval (IRB only supervises *research*)
  - But we have a collection of approved studies that you might like to do!
  - If you want to do these, you must complete CITI training (free, but will take a few hours)

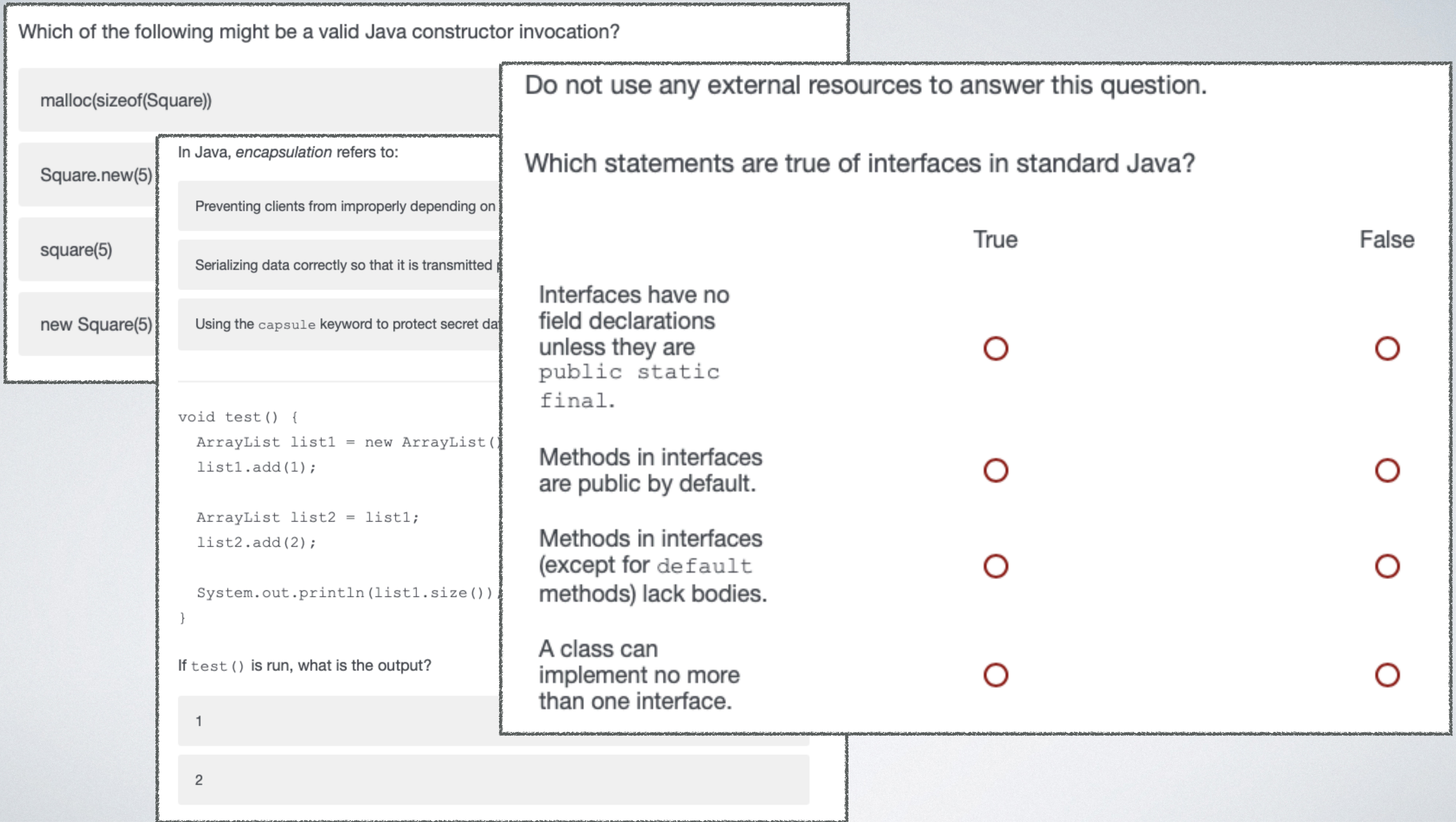
https://about.citiprogram.org/

ETHICS

- What if incentive is too high?
- What if incentive is too low?
  - IRB reviews incentives
- What if recruitment is misleading?
  - IRB reviews recruitment materials

PARTICIPANT PRE-SCREENING

- Can issue a pre-test to avoid wasting time on unqualified participants.
- Issues:
    - How will you incentivize people to take the test?
    - Can you use the test results in your research?



Which of the following might be a valid Java constructor invocation?

- malloc(sizeof(Square))
- Square.new(5)
- square(5)
- new Square(5)

In Java, *encapsulation* refers to:

- Preventing clients from improperly depending on
- Serializing data correctly so that it is transmitted
- Using the `capsule` keyword to protect secret da

```
void test() {
  ArrayList list1 = new ArrayList()
  list1.add(1);

  ArrayList list2 = list1;
  list2.add(2);

  System.out.println(list1.size())
}
```

If `test()` is run, what is the output?

- 1
- 2

Do not use any external resources to answer this question.

Which statements are true of interfaces in standard Java?

	True	False
Interfaces have no field declarations unless they are `public static final`.	○	○
Methods in interfaces are public by default.	○	○
Methods in interfaces (except for `default` methods) lack bodies.	○	○
A class can implement no more than one interface.	○	○

INFORMED CONSENT

- Disclosure of information (purpose of study, procedures, risks, benefits, compensation, data usage)

- Competency of the patient (or surrogate) to make a decision,

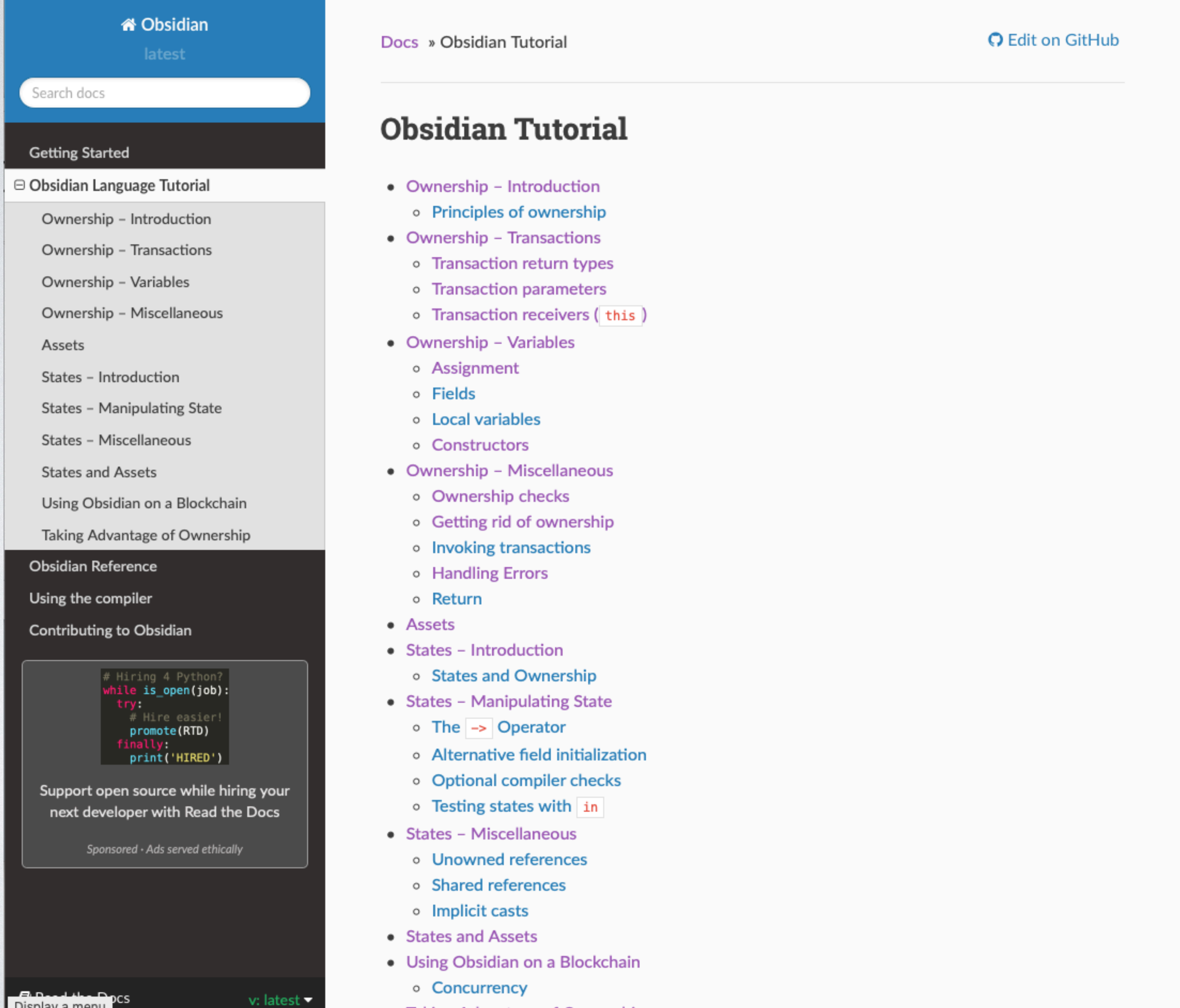
- Voluntary nature of the decision.

DEMOGRAPHICS

- Collect information if you want it!
- Programming experience? Languages?
- If they tell you, you can use it…
- e.g. Gender____________

TRAINING

- How will you prepare your participants?
- People don't read.
- People think they understand but in fact do not.
- Teach…and then assess.
- Or: decide that no training is necessary.

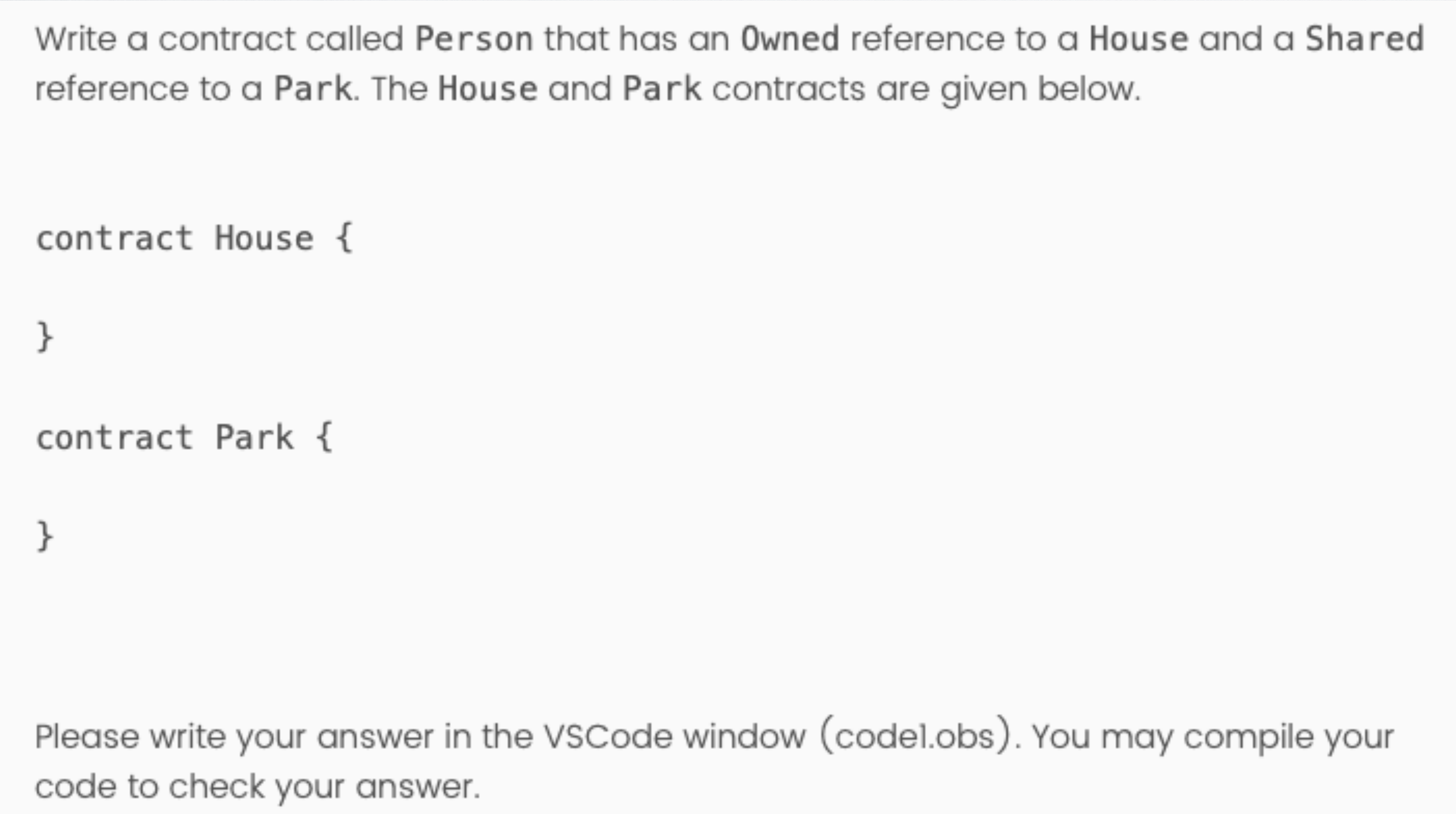


Getting Started
⊟ Obsidian Language Tutorial
Ownership – Introduction
Ownership – Transactions
Ownership – Variables
Ownership – Miscellaneous
Assets
States – Introduction
States – Manipulating State
States – Miscellaneous
States and Assets
Using Obsidian on a Blockchain
Taking Advantage of Ownership
Obsidian Reference
Using the compiler
Contributing to Obsidian

Read the Docs v: latest
Display a menu

Docs » Obsidian Tutorial   Edit on GitHub

Obsidian Tutorial



Write a contract called **Person** that has an **Owned** reference to a **House** and a **Shared** reference to a **Park**. The **House** and **Park** contracts are given below.

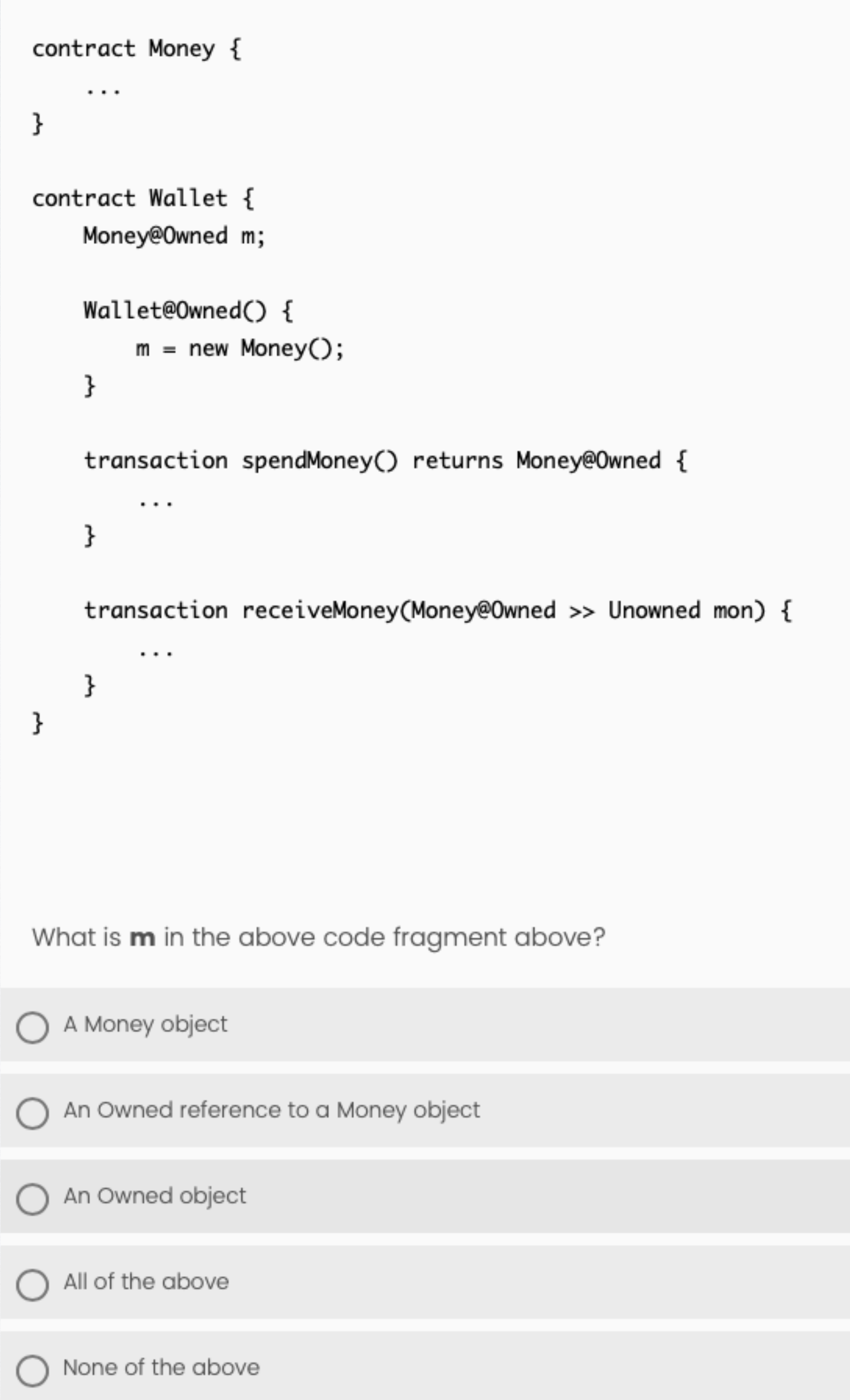
```
contract House {

}

contract Park {

}
```

Please write your answer in the VSCode window (code1.obs). You may compile your code to check your answer.



```
contract Money {
    ...
}

contract Wallet {
    Money@Owned m;

    Wallet@Owned() {
        m = new Money();
    }

    transaction spendMoney() returns Money@Owned {
        ...
    }

    transaction receiveMoney(Money@Owned >> Unowned mon) {
        ...
    }
}
```

What is **m** in the above code fragment above?

○ A Money object

○ An Owned reference to a Money object

○ An Owned object

○ All of the above

○ None of the above

TASKS

- This is the hardest part of study design.
- You will not get this right the first time.
- Solution: pilot repeatedly.
- But: you can use data from your "pilots" if you follow protocol.
- (a true "pilot" involves throwing the data out)
- What is the distribution over task times?

RECRUITMENT

- Flyers
- Emails
- Social network
- Buy ads
- The street

See: Report from Dagstuhl Seminar 19231
Empirical Evaluation of Secure Development
Processes

INCENTIVES

- $$$ (in person, MTurk)
- Desire to contribute to science / help you out
- Food
- Fame (leaderboard)
- Rare experience
- Learning opportunity
- Distraction from work
- Credit

THINK-ALOUD USABILITY STUDIES

- Give people tasks and observe what happens.
- NOT experiments
- NOT comparative
- Just want to see what problems people encounter.
- Follow "think-aloud" protocol

USABILITY STUDIES CAN SHOW

- Participants encountered the following problems…
- Participants were confused by…
- Only participants who knew X were able to do the task.

USABILITY STUDIES CANNOT SHOW

- My system is better than an existing system.

NEXT TIME

- We'll discuss task design (very tricky!)
- Read "Programmers Are Users Too: Human-Centered Methods for Improving Programming Tools." (inspiration for the title of this course)