Assignment Guidelines

- Do not put your name in your assignment. We want to grade anonymously.
- You will be graded on following good Pyret style and having examples that cover a variety of situations. Read the Testing and Style Guidelines for details.

Getting Help

- The Pyret documentation is accessible from the pirate button in the top left corner of code.pyret.org.
- Your friendly coaches and professor are here to help with this assignment! Check coaching hours and my office hours (posted soon if they’re not up yet).
- If you have any questions, please post on Ed Discussion. Make sure that any questions that include your code are marked as “staff only”.

Introduction

Giselle is the owner of a sporting goods store in Oatman, Arizona. Recently she’s seen advertisements for a competitor all over town. Afraid of losing business, Giselle hires the Oatman Ad Agency for help.

The Oatman Ad Agency recommends targeting people living nearby who would be able to get to a store in downtown Oatman. They decide specifically to target people with sporting interests in their twenties. The Ad Agency already has data on locals’ age, hobbies, town, and whether they have a car.

Task: Copy and paste these definitions into the top of your code:

```
TARGET-AGE = 25
TARGET-TOWN = "oatman"
TARGET-HOBBIES = "running, biking, walking, swimming"
```

Let’s help the ad agency figure out what customers fit Giselle’s criteria. We want to write a function that takes a person’s age, hobby, town, and car status and determines whether or not
Giselle should send them an ad. To do this, we first need to figure out some more basic information about them.

Part 1: Targeted Ads

Criteria Functions

To decide if someone is a good potential customer to advertise to, we’ll write a series of predicates – functions that return the Boolean values true or false.

1. Let’s start by writing a function to see if any potential customer is within five years of Giselle’s target age.

   Task: Define the Function:

   ```haskell
   fun within-5(age :: Number) -> Boolean:
   
   It should return true if the input, age, is within five years of TARGET-AGE and false if not. This should be an inclusive comparison, meaning that if the target age is exactly five years older or younger than the target, the function should return true.
   ```

2. Since Giselle is looking for a variety of interests that customers might have, we want to see if a customer’s interest is one of her target hobbies.

   Task: Define the Function:

   ```haskell
   fun hobby-relates(hobby :: String) -> Boolean:
   
   It should return true when the input hobby is contained in TARGET-HOBBIES and false otherwise.
   ```

3. Giselle wants to be able to see if a person’s town is within the general area of her store. However, TARGET-TOWN only gives us a specific place! To solve this issue, we need to write a function to check if a certain location is in the target town or its surrounding area.

   Task: Define the Function:

   ```haskell
   fun in-target-area(town :: String) -> Boolean:
   
   It should return true if the input is "kingman", "needles", "oatman", or "topock" (the towns around her store), and false otherwise.
   ```
4. Now we can find out if someone lives in the area, but Giselle knows that people can only get to her store if they live in the target town itself or if they can drive to the target town from another town in the area.

Task: Define the Function:

```haskell
fun in-range(town :: String, has-car :: Boolean) -> Boolean:
```

The inputs represent a customer’s town and whether they have a car or not. The function should return true if the place is TARGET-TOWN or if the input town is in the target area and the person also has a car. Otherwise, the function should return false.

Combining Criteria

Now it’s time to put all the criteria together to determine whether an ad should be shown to a potential customer or not!

Task: Define the Function:

```haskell
fun show-ad(age :: Number, town :: String, hobby :: String, has-car :: Boolean) -> Boolean:
```

The inputs are information about a potential customer, and the output is true when their age is within five years of the target age (inclusive), their town is the TARGET-TOWN or they have a car and their town is within the area, and their hobby is contained in TARGET-HOBBIES. Otherwise, the output is false.

Using Keywords

A different way of targeting ads is looking at keywords in the text of the advertisement rather than specific criteria like Giselle’s target definitions.

Task: Define the Function:

```haskell
fun show-ad2(ad :: String, age :: Number) -> Boolean:
```

The input ad is the text of an ad and age is a person’s age. The function should return true if any of the following conditions are met:
• The customer’s age is 35 or younger and the ad contains the word “active”,
• The customer’s age is 65 or older and the ad contains the word “healthy”, or
• The ad contains the word “sport”.

Otherwise, it should return false.

Part 2: Pyret Style Check

Remember that programs aren’t just written for computers to run; they’re also for people to read. Just as there is good style for writing an English essay, there is good style for writing a program.

Task: Take a moment to go back through the functions you’ve written:

• Did you write a docstring for each function?
• Do you have examples that demonstrate the functions work as intended?

If not, do so now!

Do any of your functions look like this?

```py
if some-question:
    true
else:
    false
end
```

This is very common, especially if you’ve used other programming languages before. However, the if–else is redundant; you could just write some-question and get the same result!

Task: Remove any redundant if expressions!

Part 3: Comparing Approaches

Let’s think a bit about the two approaches we’ve implemented for matching ads to people, show-ad and show-ad2.

Task: Respond to the following questions:
• How do you think these functions differ from how real ads are targeted?
• Think about the way we set up the code and the programming operations that we used:
  What are the limitations of our current code/operations for evaluating ads? What would
  you want to be able to do in code to do a better job of targeting ads?

You can put your answers in a multi-line comment at the bottom of your program

#|

Like this one!

|#

Task: In a check block – which is just like a where block that isn’t attached to a particular
function – write at least two examples that demonstrate weaknesses of show-ad and at least
two examples that demonstrate weaknesses of show-ad2.

The output from each function for each input should be surprising given the input. For
example, an ad could seem perfect for the customer, but the function would output false
because of a limitation of its design.

Part 4: Personal Data and Privacy

Beyond teaching you technical skills around computing and data, CMPU 101 also wants
to help you think about the broader societal issues surrounding them.

Task: Read the article These Ads Think They Know You on the kinds of information that
advertisers are able to access and use when targeting ads to users.

You have free access to the New York Times as a Vassar student. Follow these directions
to set up your account!
Task: Answer the following prompts using your own Google ad-matching information or – to protect your privacy – the sample shown above (from a real person). Your answer can go in a multiline comment at the bottom of your program.

1. Information that a search engine has on you comes from a combination of
   o your profile,
   o your browsing history, and
   o assumptions that the search engine makes about you based on your browsing history.

   Identify one piece of ad-matching information of each type. Your answer should look like “X likely comes from the profile, Y likely comes from browsing history, etc.”

2. Pick one piece of ad-matching information that could have come from multiple sources. Give 2–3 concrete examples of online activities (e.g., “searched for X”) that might have associated that ad-matching label with the account.

3. Pick two pieces of ad-matching information, which can be the same or different from what you used in the last question. For each, describe
   o one possible use which is beneficial or benign, and
   o one which you believe is problematic, unethical, or harmful.

Reference at least two of the following criteria somewhere in your response:

   o Intent/purpose (academic research vs. targeted ads)
   o If the data is public, does that mean it can be used in any way?
   o Timing (e.g., data set from a decade ago)
• Attributes collected (e.g., those protected by law from discrimination such as religion, race, age, or disability)
• Inference potential: What might this information help companies infer about you?

There’s no right or wrong answer here. Our goal is to get you thinking about the context of the technical content of the course. Your answer should be clear and concise, with enough specifics to show that you are thinking about the question beyond a surface level.

Submit Assignment

1. Download your file (File → Download) and ensure it’s named asmt2.arr.
2. Upload your assignment on GradeScope

Part 1 will be automatically graded when you submit your assignment, but your final assignment grade will be determined by your instructor when they review your work, including checking the programming style (Part 2) and reading your answers for Part 3 and Part 4.

You can submit as many times as you want before the deadline. Only your latest submission will be graded.

Acknowledgments

This assignment is adapted from Kathi Fisler and colleagues at Brown University, and Jonathan Gordon, Marc Smith and Jason Waterman.