

< Previous	Assignment 1	Assignment 2	Assignment 3	Assignment 4	Assignment 5	Next >
------------	--------------	--------------	--------------	--------------	--------------	--------

Assignment 3

[Bookmark this page](#)

Assignment 3

Due date: Wednesday, February 2nd, 5:00 pm Eastern Time

Coverage: Module 04

Files to submit: `a03q1.rkt`, `a03q2.rkt`, `a03q3.rkt`

Assignment Guidelines

- Language Level
- Beginning Student
- General Expectations
- Remember to use defined constants and helper functions where appropriate.
 - Follow the full design recipe as outlined below in Question 0.
- Allowable Racket Functions and Special Forms
- `define`, `and`, `or`, `not`, `cond`, `else`, `check-expect`, `check-within`
 - any type predicate such as `number?`, `boolean?`
 - any functions on numbers found in [Section 1.5 of the Racket documentation](#)
 - any functions on symbols found in [Section 1.7 of the Racket documentation](#)
 - any functions on strings found in [Section 1.11 of the Racket documentation](#)
- Other Information
- Look through the list of allowable built-in functions linked from above. You might find some handy functions that were not explicitly mentioned in the modules!
 - All test data for correctness will always meet the stated assumptions for consumed values.

Question 0: Style Guide

This is the first assignment requiring full use of the Design Recipe. Read up to and including Section 3.6.1 of the [Coding Style](#) document. Your submission for this and subsequent assignments must adhere to the recommendations in that document. A significant number of marks will be allotted to each of contracts, purposes, examples, definitions and tests. Also, as in previous assignments, you may lose marks for poorly styled code (e.g., inappropriate names for constants and functions or improper spacing and indentation).

There is nothing to submit for this question.

Question 1: Pig Latin

Pig Latin is a specific way of rearranging letters in English words for fun.

Vowels ('a', 'e', 'i', 'o', and 'u') are treated separately from the consonants (any letter that isn't a vowel). For simplicity, we will consider 'y' to always be a consonant.

Although various forms of Pig Latin exist, we will use the following rules:

- Words of two letters or less simply have "way" added on the end. So "go" becomes "goway".
- In any word of three or more letters that starts with consonants, the consonants are moved to the end, and "ay" is added. But if it begins with more than two consonants, move only the first two letters to the end. So "hello" becomes "ellohay", and "string" becomes "ringstay".
- Any word which begins with a vowel simply has "way" added on the end. So "explain" becomes "explainway".

Write a function `pig-latin`. It consumes a string, and produces that string converted to Pig Latin, following the rules described above. The string shall contain only lowercase letters and have a length of at least 1.

For example:

```
> (pig-latin "this")
"isthay"
> (pig-latin "is")
"isway"
> (pig-latin "a")
"away"
> (pig-latin "crazy")
"azycray"
> (pig-latin "exercise")
"exerciseway"
```

Submit your solution in the file `a03q1.rkt`.

Question 2: Currency Encoding

In order to make it easier to read numbers, we often break a number into powers of one thousand, and add a separator (such as a comma or space) between these groups. For example, 10^8 is easier to read when written 100,000,000 than when written 100000000.

In addition, when encoding prices, many currencies (including \$, €, £, and at least historically, ₤) are divisible into 100 smaller pieces. This smaller piece is often called a "cent" from the Latin word for hundred.

Write a function `encode-currency`. It consumes a number `price` and produces a string representing this number. As needed, the string should use a comma ", " as a thousands separator, and period "." as a decimal point. You may assume that `price` is non-negative and less than one million, and does not have more than 2 decimal places.

- If a decimal point is used, the number must be represented with 2 decimal places.
- If the number can be expressed exactly without cents, the string should not include a decimal point.

For example,

```
> (encode-currency 42)
"42"
> (encode-currency 42.5)
"42.50"
> (encode-currency 420000)
"420,000"
```

Save your definitions in a file called `a03q2.rkt` and submit it to MarkUs.

Question 3: Currency Decoding

This question is the "reverse" of the previous question, and uses the same conventions.

In addition, some countries use different conventions for thousands separators and decimal places. In much of the world, it is common to use a period ".", " as the thousands separator, and a comma ", " as the decimal point. We want to be able to work with data written in either format.

Write a function `decode-currency`. It consumes a string representing a price less than one million, either using comma as the thousands separator and period as the decimal place, or using period as the thousands separator, and comma as the decimal place. The function produces the corresponding number. For example:

```
> (decode-currency "42")
42
> (decode-currency "2.50")
2.5
> (decode-currency "2,19")
2.19
> (decode-currency "21,000.45")
21000.45
> (decode-currency "765.432,10")
765432.1
```

Save your definitions in a file called `a03q3.rkt` and submit it to MarkUs.

Discussion

Topic: Assignments / Assignment 3

Hide Discussion

Add a Post

Show all posts	by recent activity
<div><div>?</div><div>Q3</div><div>Hi, I am aware that the assignment deadline has passed, however a question did come to mind about question 3 of the assignment. Since decimals and commas can be intercha...</div><div>2</div></div>	
<div><div>?</div><div>Contract</div><div>Hi, I noticed in module 4 section 4.4 when a "Requires:" was used, and there were more than one requirement, there was a use of * before every requirement. However, when I c...</div><div>2</div></div>	
<div><div>💬</div><div>Whitespace Marking</div><div>Would it be possible to receive more guidelines about proper whitespace and indentation etiquette? After reviewing my marks of assignment 2 I lost several marks on illogical w...</div><div>2</div></div>	
<div><div>?</div><div>Allowable functions</div><div>Hi, Can we use (equal?) ? Thanks,</div><div>2</div></div>	
<div><div>?</div><div>Q3</div><div>Hi, Just some clarification for question three. The question states that Q3 uses the same conventions as Q2, however in the example (decode-currency "765.432,10"). We see h...</div><div>2</div></div>	
<div><div>?</div><div>Question for Q1</div><div>is there any other efficient way to check whether or not a part of the string is a consonant/vowel other than using "equal?" and going through all the different consonants and v...</div><div>2</div></div>	
<div><div>?</div><div>Question 2</div><div>Should we consider cases where the decimal value is zero, or in this case two zeroes? ex. 45.00</div><div>2</div></div>	
<div><div>?</div><div>Question 3</div><div>In order to convert the string to a number, I'm using the $str \in g \rightarrow number$ function. However, this only works for integers. Otherwise, it returns it as a mixed fraction. Thus, I then u...</div><div>4</div></div>	
<div><div>?</div><div>Question 3</div><div>Hi, I was just wondering if we can assume that the string value will have just 2 decimal places?</div><div>4</div></div>	
<div><div>?</div><div>Q2</div><div>Do we have to account for the possibility of decimal places **and** commas like in question three? so 10000.5 would turn into 10,000.50 in question two?</div><div>2</div></div>	
<div><div>?</div><div>Q3 Separator Clarification</div><div>Should our code be able to convert numbers in the thousands with the decimal separator and no cents? For example, 12000 expressed as 12.000. Or will we only see this conve...</div><div>3</div></div>	
<div><div>?</div><div>Allowed Functions</div><div>Are we allowed to use functions that utilize characters? For example, I want to use the built-in function string-ref and it produces a character type. It is in Section 1.11 which is o...</div><div>3</div></div>	
<div><div>?</div><div>How many examples we need?</div><div>Hi I'm wondering how many examples we need for each question? Would 2 examples be enough?</div><div>2</div></div>	
<div><div>?</div><div>Question 2</div><div></div><div></div></div>	