

Discussion 3: Domain, Range, Lists and HOFs

SOLUTIONS

Domain and Range

1. Determine the domain and range of the following Snap! blocks:



Domain: **First blank: list, second blank: any value**

Range: **Booleans (True/False)**






Domain of foo: **Numbers**

Range of foo: **Booleans (True/False)**

Data type of var: **Booleans (True/False), since the type of var is the same as the output type of foo**

2. Fill in the table with the domain and range of the following higher order functions:

Higher Order Function	Domain	Range	Notes
 The 'map over' block is an orange block with the word 'map' on the left, a small circle icon, and the word 'over' on the right with a small square icon.	<p>First blank: reporter, with zero or more of its input slots left blank</p> <p>Second blank: list</p>	<p>List with same length as input list</p>	<p>~ The reporter must be able to take in all the data types in the input list without erroring. For example, if the input list to map has numbers and words, the input reporter must be able to handle numbers and words.</p> <p>~ The output type of the reporter determines the data types of the values in the output list. For example, if the reporter outputs Booleans, then the output list will only have Boolean values.</p>

	<p>First blank: predicate, zero or more of its input slots left blank Second blank: list</p>	<p>List with length less than or equal to length of input list</p>	<p>~ The domain of the predicate must include ALL data types in the input list. For example, if the list contains booleans and numbers, the predicate must be able to handle both booleans and numbers. ~ Keep should never modify the items in the input list when creating its output list. Every item in the list outputted by keep MUST also be in the input list</p>
	<p>First blank: reporter with two input slots left blank Second blank: list</p>	<p>A single value of any data type</p>	<p>~The type of the output value is the same as the output type of the reporter. For example, if the reporter outputs numbers, combine will output a number. ~The input reporter must be able to handle all data types from the input list, as in map and keep.</p>

Higher Order Function Practice

1. Fill in the blanks so the keep block returns a list of the numbers from MyList.

```
MyList = [3, hello, goodbye, 5, 6]
```

Keep items such that  from MyList

2. Write an expression that returns the sum of the squares of the numbers in YourList.

YourList = [1, 2, 3, 4]

```
combine with (+) items of map (x) over YourList
```

3. Complete the following block so it works as described. Note: You may find the sentence -> list block helpful.

```
prepend every anti [dote·pasto·gone·body] antidote antipasto antigone antibody
+prepend+every+ word + sentence +
+prepend+every+ word + sentence +
report combine with join [ ] items of
map join word [ ] over sentence → list sentence
```

4. Describe (in words) what the following block outputs. Assume OurList is a list of words.

```
combine with join [ ] items of
map letter length of [ ] of [ ] over OurList
```

It outputs a word consisting of the last letter of each word from our list.

Challenge Problems

1. Determine the domain and range of the following blocks:

a.

Domain of Foo: **Booleans**, since the output of the equals block is passed in

Range of Foo: **A list of numbers**, since we know that the output of Foo is passed into the map block

a.

Domain of Bar: **Not enough information**

Range of Bar: **Numbers**, since the output of Bar can be passed into the ">" block and can be compared with a number

2. If the output of Mystery is true, which of the following can you say for sure?

- | | | | | |
|--------------------------|--------------------------|-------------------------------------|-------------------------------------|--------------------------|
| <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| A must be true | B must be true | A must be false | B must be false | None of these |

If Mystery outputs true, then it could not have gone into either of the if statements, so A and B must be false.

```

+Mystery+ A + B +
if A
  report false
if B
  report false
report true
  
```

3. You realize you could replace the *entire* body of Mystery with a single report statement. What could we report instead so that Mystery would have the same exact behavior? (Select all that apply)

- A and B
- A or B
- not A and not B
- not A or not B
- not A or B
- not A and B

To get this answer, we test each of the blocks with every possible combination of A and B (TT, TF, FT, FF) and find the ones that match the behavior of Mystery.

4. Given the following expression, what does NAMES evaluate to?

```

set NUMBERS to list 2 1
set NAMES to list Dan Garcia
for i = 1 to 2
  replace item item i of NUMBERS of NAMES with i
  
```

a.

b.

c.

d.

5. One of the most common data storage technologies is databases, think of them as Tables/Charts, with columns and rows. Let's say you're given a table that looks like the following:

ID	Name	Height(inches)	Year	2 Favorite Numbers
1	Dan	75	4	[10, 61]
2	Mansi	65	4	[161, 10]
3	Bob	70	3	[70, 170]
4	Alice	71	1	[10, 160]
5	Nick	68	3	[161, 162]
6	Eve	64	2	[270, 370]
:	:	:	:	:
:	:	:	:	:
...

Note: Each entry in "2 Favorite Numbers" is a list with exactly 2 numbers!

6	A	B
1	10	61
2	161	10
3	70	170
4	10	160
5	161	162
6	270	370

You're given a reporter block, **Column**, that takes in the name of a column in the database and reports a list of all of the elements in that column in order. An example call is shown to the left.

Column 2FavoriteNumbers

For this problem, you can assume that you are only given the 3 HOFs, **Column**, and

any operators (green blocks) in Snap! For each of the subparts before, determine if the problem can be solved using only the given blocks.

a Report the total sum of heights **Yes** No

Yes, we can use combine on Column(height)

b Report a list of the names of people who are in year 4 **Yes** **No**

No, our higher order functions cannot take in multiple lists. It can tell which years are equal to 4, but not which rows those 4s came from

c Report the number you get when you multiply the squares of all of the heights above 70 inches together. **Yes** No

The image shows a Snap! code block for 'combine with' with a multiplication operator (x) selected. It contains three sub-blocks: 'items of', 'map over', and 'keep items such that'. The 'keep items such that' block has a greater-than operator (>) and the number 70, and is connected to a 'Column Height' block.

d Calculate the sum of all of the numbers in the "2 Favorite Numbers" column **Yes** No

```

combine with (+) items of
map (combine with (+) items of) over
Column 2 Favorite Numbers

```

6. Indicate whether each set of blocks below is equivalent:

a

```

map (+ 2) over
keep items such that (> 3) from list 2 3 4 4

```

No, the first block outputs [6, 6], and the second block outputs [4, 5, 6, 6]

```

keep items such that (> 3) from
map (+ 2) over list 2 3 4 4

```

b

```

map (+ 3) over map (x 2) over list 2 3
map (x 2) over map (+ 3) over list 2 3

```

No, the first block outputs [7, 9], and the second block outputs [4, 12]

c

```

map (+ 3) over map (+ 6) over list 2 3
map (+ 6) over map (+ 3) over list 2 3

```

Yes! Both output [11, 12]

d

```

keep items such that (not not) from
list false true
keep items such that (not) from
not list false true

```

No, this is a trick question! The second block will cause an error, because not cannot take in a list.