Final Project: Milestone 3

CS 327E

April 9, 2018

Announcements:

Next Week: Last regular class
Demo Day: Friday 04/27
Demo Location: WAG 420
Demo Schedule: https://tinyurl.com/yd68gutt

1) Which of the following are the **core** computation and storage components of Hadoop?

A) Pig and Hive
B) Spark and YARN
C) MapReduce and HDFS
D) All of the above.

2) The basic MapReduce programming model consists of which types of operations?

A) A Map function, supplied by the user.
B) A Reduce function, supplied by the user.
C) An optional Combiner function, supplied by the user.
D) All of the above.

3) Which of the following is **not** an example of a key-value pair record?

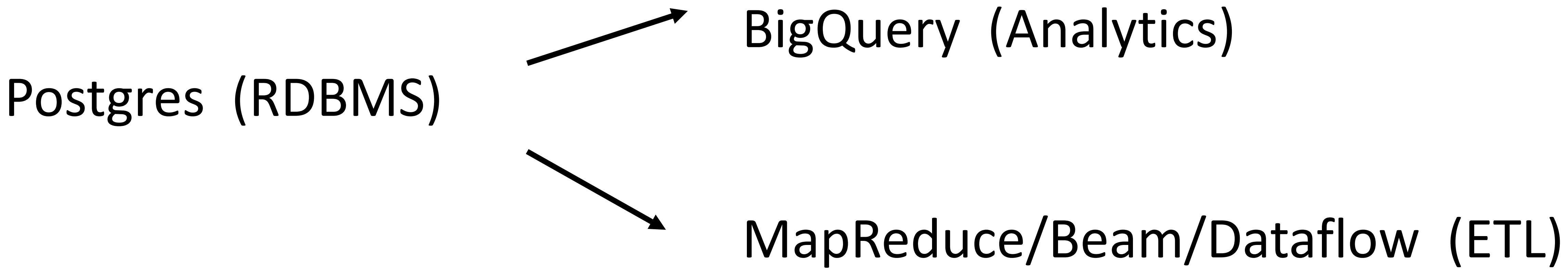
A) ('http://www.utexas.edu', 'utexas.edu')
B) ('The', 929)
C) ('The', 929, '04-09-2018')
D) ('kinglear.txt', 'Captains, Messengers, Soldiers, and Attendants…')

4) What is a key property of the **shuffle** procedure?

A) The map workers receive a split of the input data.
B) The reduce workers receive all the values that share the same key.
C) The distributed file system uses 3-way replication.
D) All of the above

5) What kind of failures can the MapReduce system tolerate?

A) Map worker failures
B) Reduce worker failures
C) Job Tracker / Master failures
D) Disk failures
E) A, B, D

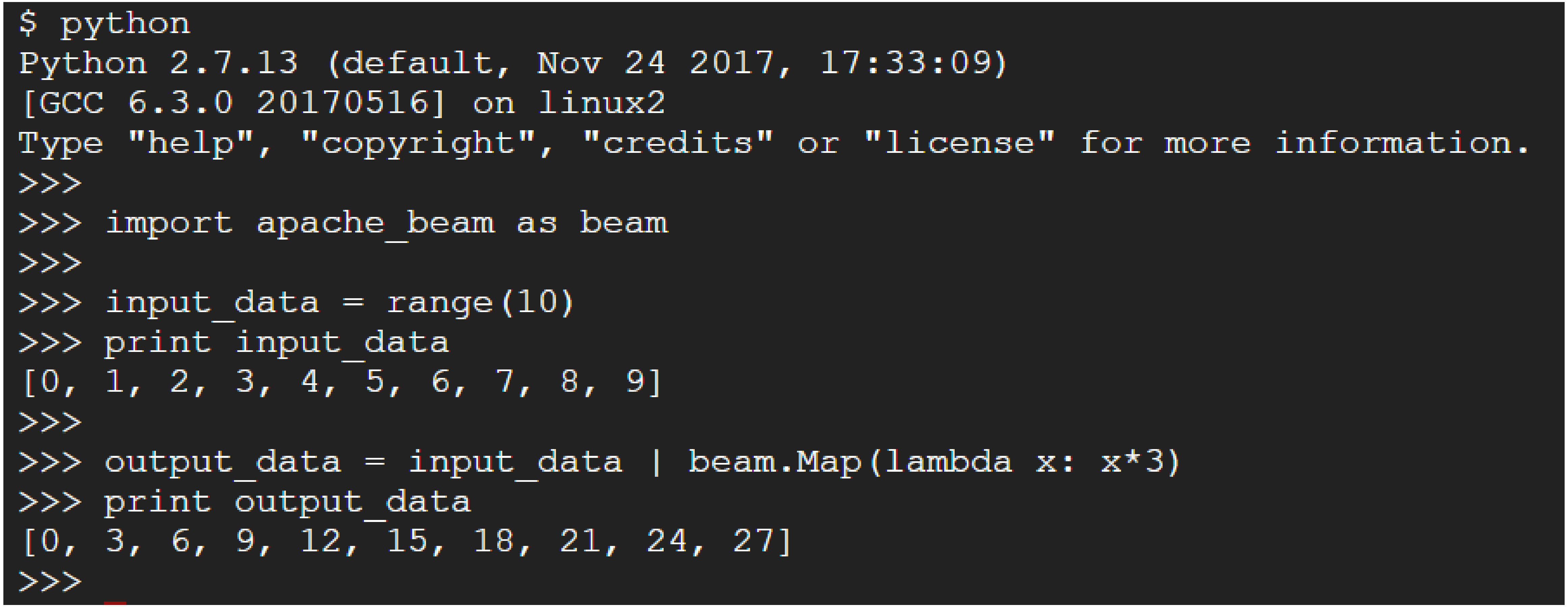


Postgres (RDBMS)

BigQuery (Analytics)

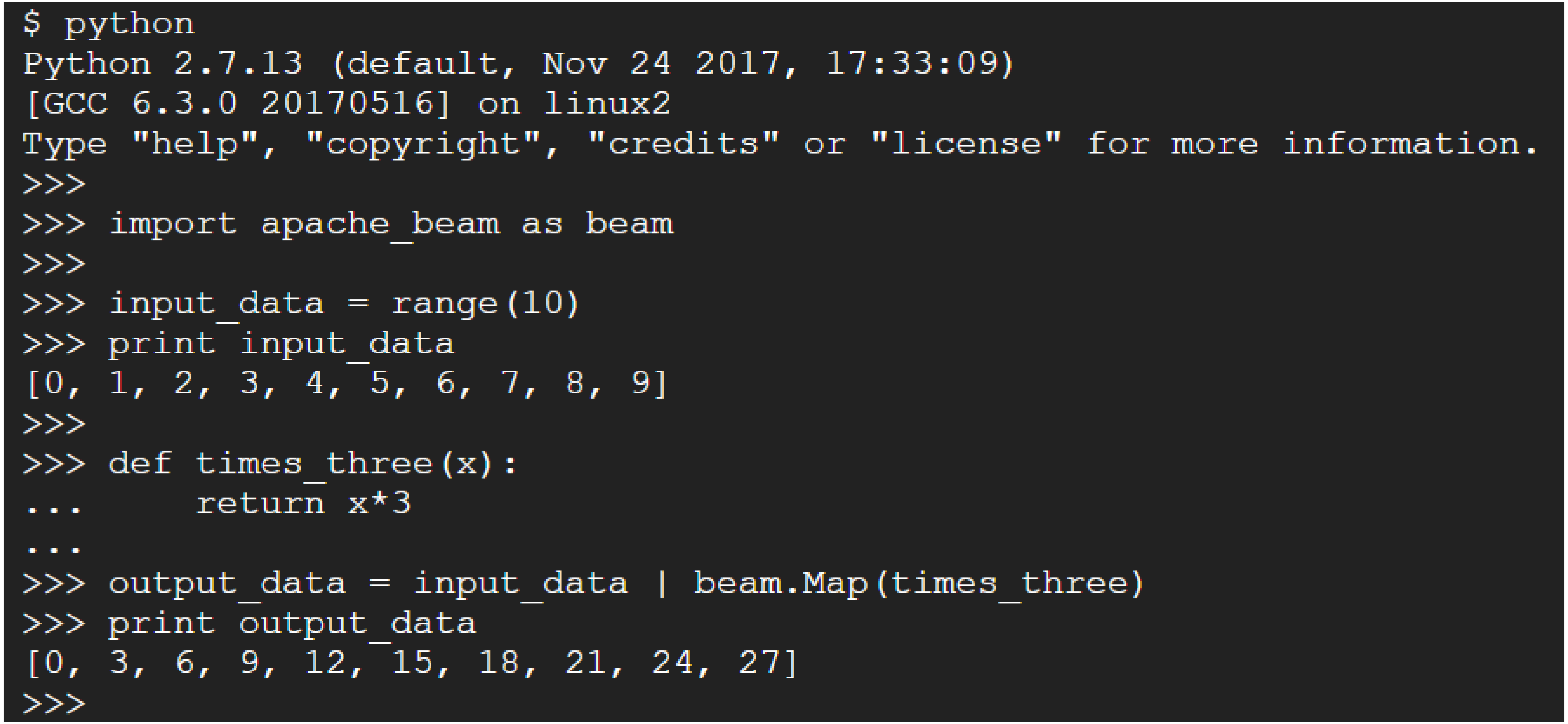
MapReduce/Beam/Dataflow (ETL)

Example Map Function v1



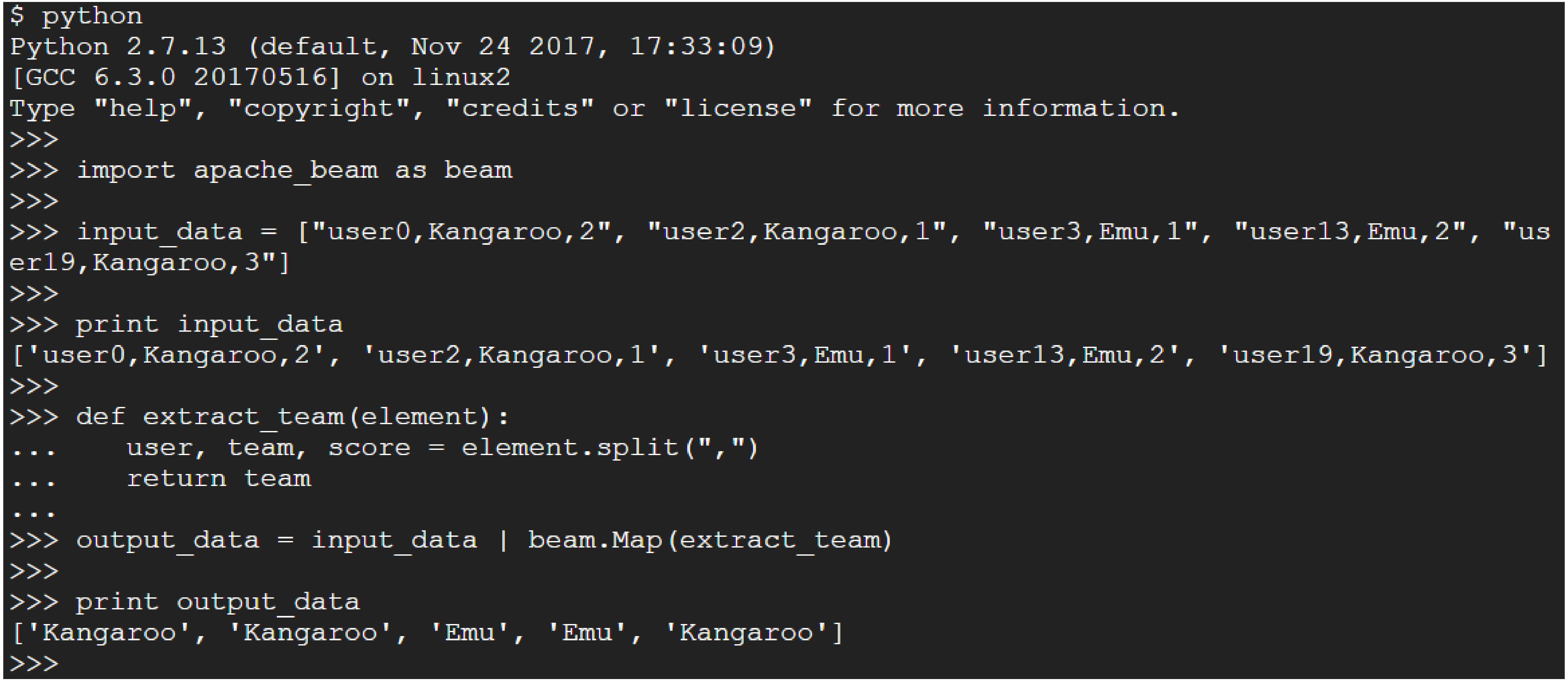
```
$ python
Python 2.7.13 (default, Nov 24 2017, 17:33:09)
[GCC 6.3.0 20170516] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> import apache_beam as beam
>>>
>>> input_data = range(10)
>>> print input_data
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
>>>
>>> output_data = input_data | beam.Map(lambda x: x*3)
>>> print output_data
[0, 3, 6, 9, 12, 15, 18, 21, 24, 27]
>>>
```

Example Map Function v2



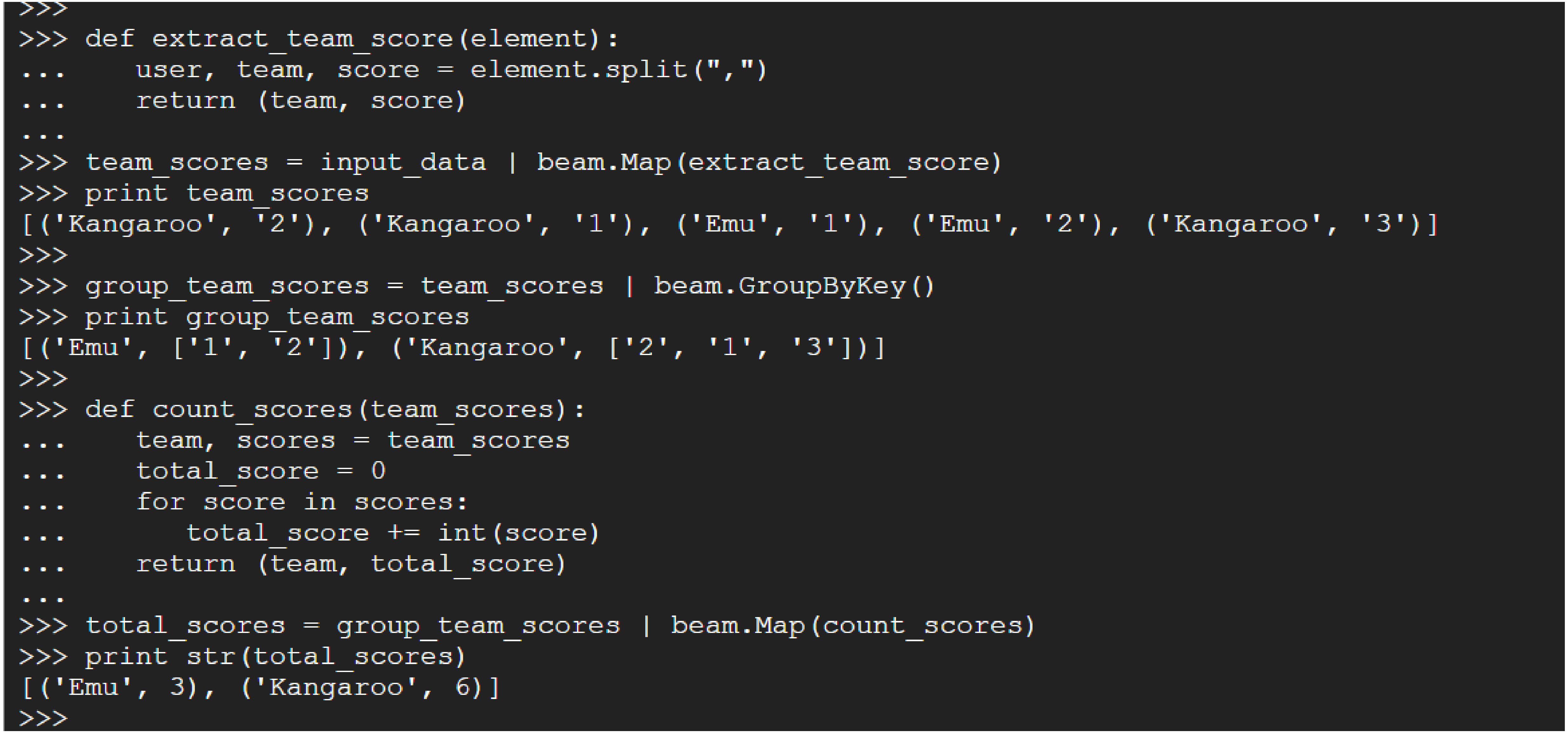
```
$ python
Python 2.7.13 (default, Nov 24 2017, 17:33:09)
[GCC 6.3.0 20170516] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> import apache_beam as beam
>>>
>>> input_data = range(10)
>>> print input_data
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
>>>
>>> def times_three(x):
...     return x*3
...
>>> output_data = input_data | beam.Map(times_three)
>>> print output_data
[0, 3, 6, 9, 12, 15, 18, 21, 24, 27]
>>>
```

Another Example Map Function



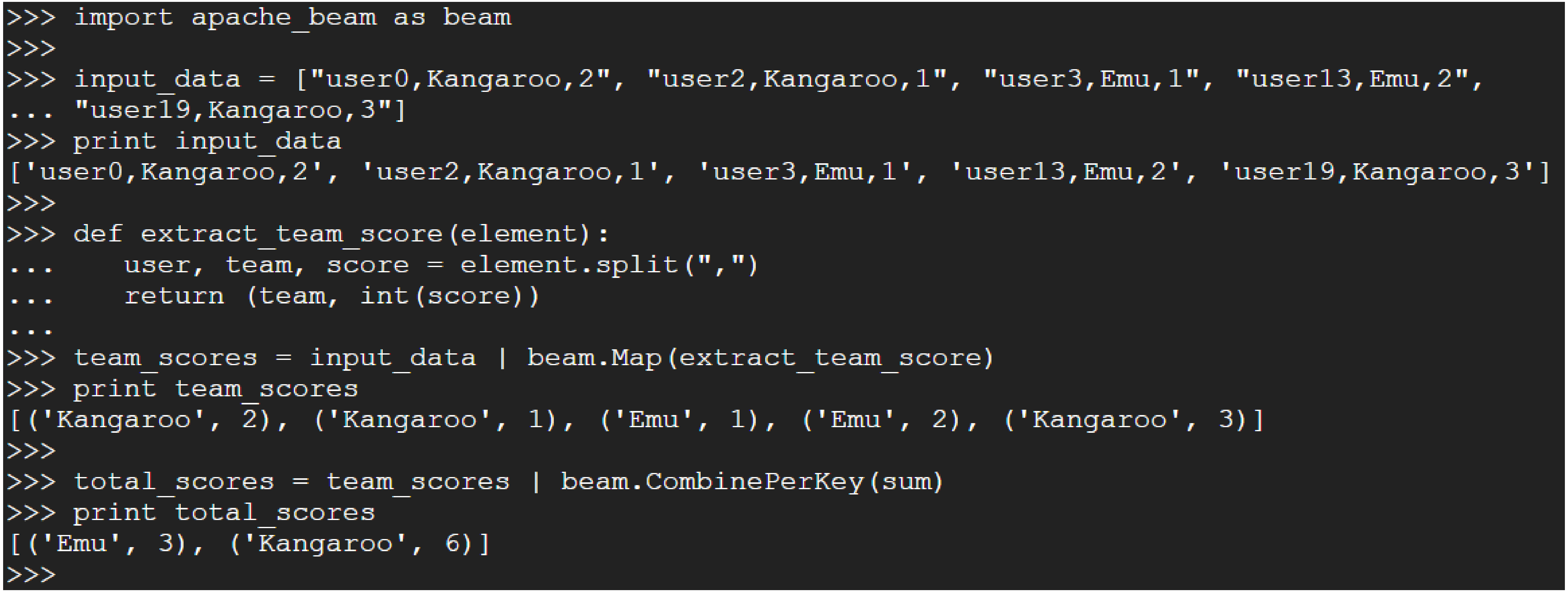
```
$ python
Python 2.7.13 (default, Nov 24 2017, 17:33:09)
[GCC 6.3.0 20170516] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> import apache_beam as beam
>>>
>>> input_data = ["user0,Kangaroo,2", "user2,Kangaroo,1", "user3,Emu,1", "user13,Emu,2", "us
er19,Kangaroo,3"]
>>>
>>> print input_data
['user0,Kangaroo,2', 'user2,Kangaroo,1', 'user3,Emu,1', 'user13,Emu,2', 'user19,Kangaroo,3']
>>>
>>> def extract_team(element):
...     user, team, score = element.split(",")
...     return team
...
>>> output_data = input_data | beam.Map(extract_team)
>>>
>>> print output_data
['Kangaroo', 'Kangaroo', 'Emu', 'Emu', 'Kangaroo']
>>>
```

Example Map and GroupByKey Functions



```
>>>
>>> def extract_team_score(element):
...     user, team, score = element.split(",")
...     return (team, score)
...
>>> team_scores = input_data | beam.Map(extract_team_score)
>>> print team_scores
[('Kangaroo', '2'), ('Kangaroo', '1'), ('Emu', '1'), ('Emu', '2'), ('Kangaroo', '3')]
>>>
>>> group_team_scores = team_scores | beam.GroupByKey()
>>> print group_team_scores
[('Emu', ['1', '2']), ('Kangaroo', ['2', '1', '3'])]
>>>
>>> def count_scores(team_scores):
...     team, scores = team_scores
...     total_score = 0
...     for score in scores:
...        total_score += int(score)
...     return (team, total_score)
...
>>> total_scores = group_team_scores | beam.Map(count_scores)
>>> print str(total_scores)
[('Emu', 3), ('Kangaroo', 6)]
>>>
```

Example Map and CombinePerKey Functions



```
>>> import apache_beam as beam
>>>
>>> input_data = ["user0,Kangaroo,2", "user2,Kangaroo,1", "user3,Emu,1", "user13,Emu,2",
... "user19,Kangaroo,3"]
>>> print input_data
['user0,Kangaroo,2', 'user2,Kangaroo,1', 'user3,Emu,1', 'user13,Emu,2', 'user19,Kangaroo,3']
>>>
>>> def extract_team_score(element):
...     user, team, score = element.split(",")
...     return (team, int(score))
...
>>> team_scores = input_data | beam.Map(extract_team_score)
>>> print team_scores
[('Kangaroo', 2), ('Kangaroo', 1), ('Emu', 1), ('Emu', 2), ('Kangaroo', 3)]
>>>
>>> total_scores = team_scores | beam.CombinePerKey(sum)
>>> print total_scores
[('Emu', 3), ('Kangaroo', 6)]
>>>
```

Final Project Milestone 3

Beam/Dataflow Job Requirements:
http://www.cs.utexas.edu/~scohen/project/fp_guidelines.pdf

Dataflow Setup Procedure: https://github.com/cs327e-spring2018/snippets/wiki/Dataflow-Setup-Guide

Beam Code Samples (beam1.py – beam5.py):
https://github.com/cs327e-spring2018/snippets/tree/master/beam