Dagskrá í viku 7: Hlutbundin forritun og einstaklingsverkefni 2

- Miðvikudagurinn 18.2
  - Python upprifjun
  - Kynning á hlutbundinni forritun
  - Meira um klasa í Python
  - Kynning á einstaklingskverkefni 2
  - Tímaverkefni

Smá Python upprifjun í boði Google

Upprifjun fyrir þá sem það þurfa, vídjókennsla:

- Hluti 1 (Introduction and Strings)
- Hluti 2 (Lists, Sorting, and Tuples)
- Hluti 3 (Dicts and Files)
- Hluti 4 (Regular expressions)
- Hluti 5 (Utilities: OS and Commands)
- Hluti 6 (Utilities: URLs and HTTP, Exceptions)
- Hluti 7 (Closing thoughts)

Hvað er hlutbundin forritun?

Object Oriented Programming

Hvað er klasi?

Classes and Self

Að búa til smiði

Create Constructors

Dæmi um klasa

```
class Employee:
	'Common base class for all employees'
	empCount = 0

	def __init__(self, name, salary):
		self.name = name
		self.salary = salary
	Employee.empCount += 1

	def displayCount(self):
		print "Total Employee %d" % Employee.empCount

	def displayEmployee(self):
		print "Name : ", self.name, ", Salary: ", self.salary
```

Sjá nánar hér:
http://www.tutorialspoint.com/python/python_classes_objects.htm

Búum til tilvik (instance) af klasanum

```
"This would create first object of Employee class“
 emp1 = Employee("Zara", 2000)

"This would create second object of Employee class"
emp2 = Employee("Manni", 5000)
```

Köllum á föll í eigu tilvikanna og gildið á breytu sem klasinn á

```
emp1.displayEmployee()
emp2.displayEmployee()

print "Total Employee %d" % Employee.empCount
```

Sem prentar út:

```
Name : Zara ,Salary: 2000        (Hér er keyrt: emp1.displayEmployee())
Name : Manni ,Salary: 5000       (Hér er keyrt: emp2.displayEmployee())

Total Employee 2                 (Hér er neðsta skipunin keyrð, klasinn veit
                                 sjálfur hvað stendur í empCount breytunni
                                 sinni og sú tala er óháð tilvikunum)
```

Fyrir þá sem vilja læra meira:

- Subclasses Superclasses
- Overwrite Variable on Sub
- Multiple Parent Classes

Einstaklingsverkefni 2

- Við ætlum að búa til einfaldan kapal án grafísks viðmóts.
- Forritið þarf að notast við amk einn klasa
- **Reglur**:

  Þið þurfið að búa til stokk sem er raðaður randomly í hvert skipti sem þið keyrið kapalinn. Svo dragið þið spil úr stokknum og þegar þið fáið tvö spil af sömu sort með tveimur spilum á milli megið þið taka þau spil í burtu sem eru á milli. Ef þið sjáið tvö spil með sama gildi og tvö á milli þá megið þið taka öll fjögur í burtu. Þegar við erum komin á þann stað að vera búin að fletta upp öllum spilum úr stokknum má taka aftasta spilið og setja það fremst og athuga hvort hægt sé að halda áfram með kapalinn þannig. Til að vinna kapalinn eiga engin spil að vera eftir á hendi eða tvö spil.
- **Dæmi**:

  Spilin mín eru: [H5] [H6] [S2] [L13] [T8] [S12] þegar síðasta spilið kemur upp þá get ég tekið [L13] og [T8] úr stokknum. Eftir það lítur hann þá svona út: [H5] [H6] [S2][S12] og við höldum áfram að draga þar sem [H5] og [S12] eru ekki af sömu sort. Ef við værum t.d. með [H5] [H6] [S2][S5] þá gæti ég fjarlægt öll spilin fjögur. Þegar komið er að síðustu spilunum má setja aftasta spilið fremst, þ.e. Spilið lengst til hægri má setja lengst til vinstri. Ef [H5] [H6] [S2] [S12] væru síðustu spilin mín þá gæti ég gert [S12] [H5] [H6] [S2] og þarna mætti ég fjarlægja [H5] og [H6] og eftir standa [S12] og [S2] og ég vann kapalinn. JEIJ! ☺

Tímaverkefni

- Hvað var erfiðast í fyrsta hópverkefninu?
- En léttast?