Lists – Day 1

10/30/18

Journal

- Students will create lists, access, and traverse elements within the list.
- Students will perform list operations including insertion, concatenation, repetition, slices, and deletion.
- Students will be introduced to list methods including append, insert, pop, sort, reverse, index, count, and remove.
- Students will explore the Python API.
- Students will use lists to model several real-world situations.

write each command and the output [2:3)

0 1 2 3

name = “Paliszewski”

1. len(name) → 11
2. name[2:3] → l
3. name.find(‘z’) → 5
4. name[:4] → Pali
5. for character in name:
   print(character)

P
a
l
i
s
z
e
w
s
k
i

CS Matters

ListyList Role Play step1

[ ]

```
students = []

students.append(‘Christian’)

students.append(‘Jason’)

students.append(‘Elias’)

students.append(‘Everett’)

students.append(‘Cesar’)

students.append(‘Liz’)
```

ListyList Role Play step 2

`print(len(students))` 6

`print students[3]` Everett

`students.reverse()`

`print(“James” in students)` False

`students.sort()`

CS Matters

ListyList Role Play step 3

```
more = [“Matthew”*2, “Gabe”]

students = students + more

del students[1]

del students[2:4]
```

ListyList Role Play step 4

```
stu = students.pop(1)

#stu2 = students.pop()

students.insert(0, ‘Jonah’)

print(students.index(‘Cesar’) )   = 1
```

Lists within a list (creating a zoo)

```
def newAnimal( )
    input at least 3 qualities
    return the name, qualities

def printZoo(animals)
    for index in animals
        print(index)

defmain( )
    zooList [ ]
    quit = "yes"
    while (quit != "no"):
        animal = newAnimal( )
        zooList.append(animal)
        quit = input("any more? yes or no")
    print(zooList)
```

CS Matters