1.

**Code:**

>>> a

**Produces:**

NameError

**Why:**

A is not a defined variable/method in the python code. Because of this the alpha character ‘a’ is meaningless to the python complier.

2.

**Code:**

>>> x = 5

>>> y = 25

>>> x + y = 500

**Produces:**

SyntaxError

**Why:**

It cannot assign the value on the right side (500) to x + y because both x and y and defined as 5 and 25 respectively. Because of this Python cannot assign the value 500 to the equivalent value of 30 (or either of the corresponding variables in the expression).

3.

**Code:**

>>> int("hi")

**Produces:**

ValueError

**Why:**

int is a method that is defined only to work on numbers to translate them into integer numbers. By putting in a string value we asked Python to perform a function on something that isn’t in the domain of that function.

4.

**Code:**

>>> list = [1,2,3,4,5]

>>> list[20]

**Produces:**

IndexError

**Why:**

We are asking Python to give us the 21st item in the list “list” however “list” only has 5 items thus Python cannot return a non-existent value.

5.

**Code:**

>>> from math import integral

**Produces:**

ImportError

**Why:**

The module “math” doesn’t contain any variables or functions defined as “integral” thus Python cannot import said module definition.

6.

**Code:**

>>> 44 / 0

**Produces:**

ZeroDivisionError

**Why:**

Because mathematically any number divided by zero isn’t defined so Python cannot divide something by zero.

7.

**Code:**

>>> while True:

print("hit me baby one more time")

hit me baby one more time

hit me baby one more time

.

.

.

(I eventually press Ctrl-c)

**Produces:**

KeyboardInterrupt

**Why:**

The program was still running and would continue to run because it was an infinite while loop and when I pressed

8.

**Code:**

>>> 1230928309832 \*\* 120938209810239820938

**Produces:**

Overflow Error

**Why:**

The number that results from this calculation is too big for Python to represent

9.

**Code:**

>>> "kansas" + 120

**Produces:**

TypeError

**Why:**

The operator + is defined in a way that you can add the two different types int and str together. If it were both str’s or both int’s we would be fine but you cannot mix and match.

10.

**Code:**

>>> dicttest = {"key1": "value1"}

>>> dicttest["key2"]

**Produces:**

KeyError

**Why:**

Much like the index error with lists for the dictionary we are asking to access a value from a key that isn't defined inside the dictionary. Because of this Python cannot return any value and thus we get an error.