School Nova Computer Science Year 1 Core Topics. Instructor: Oleg Smirnov.

#	Topic	Notes
1	Using Python editor, saving your code in a file, running your code.	
2	Arithmetic operators. Comparison operators.	
3	Logical operators: *and, or, not.*	
4	String, integer, float, Boolean: definitions and basic operations.	
5	Type conversion: str(), int(), float(), bool(). Function type().	
6	Function id(). Difference between an object and a reference to it. Python variable names.	
7	Print() and f-strings.	
8	Input() and its applications. Type conversion revisited.	
9	Error handling: try-except-else structure. Purpose and applications.	
10	Iteration (indefinite): while, continue, break.	
11	Iteration (definite): for loop. Functions range() and len().	
12	Conditional statements: if else, if elif else.	
13	Lists: format, indexing, slicing, nested list.	
14	List operations: append, extend, remove, insert, pop, del, in, not in.	
15	List copy() and deepcopy(). Difference between '=' and copy().	
16	Differences between lists, tuples, sets, and dictionaries.	
17	Type conversion 2.0: list(), tuple(), set(), dict(). Empty data structures.	
18	Set methods: union(), update(), intersection(), difference(), and so on.	
19	Dictionary: using/updating keys, accessing/adding elements.	
20	Dictionary methods: get(), items(), keys(), pop(), update(), values().	
21	Converting list and tuples to dictionaries, zip() function.	
22	Nested dictionaries and other data structures.	
23	User defined functions. Different types of arguments. Empty, default, keyworded.	
24	Nested functions. Local and global variables. Return values (strings, integers, lists).	
25	Anonymous lambda functions.	
26	map() and filter() functions. Using lambda and map() functions together.	
27	List comprehension.	
28	Object-oriented programming introduction. Terminology (class, instance, method).	
29	Class versus instance: attributes, variables, and methods. Generating multiple objects of a class.	
30	Random number generation (using import random).	
31	Different classes and polymorphism. Applications.	
32	Inheritance. Functions super(), isinstance() and issubclass().	
33	Module os. Current working directory, creating and change directories.	
34	Opening and reading data from text files.	
35	Working with text. Calculating total number of characters, words, and sentences.	
36	Import existing scripts/code. Calculating most frequent words in a text file.	
37	Modules: string and collections. string.punctuation. collections.Counter().	
38	Writing text to files. Appending text to files.	
39	Numerical Python (NumPy). Array properties (dtype, ndim, shape, size). Indexing values.	
40	Basic operations with arrays: creation, reshaping, manipulation, stacking, element-wise math.	
41	Numpy random number generation (randint, uniform). Functions np.sum(). np.mean().	
42	matplotlib. Creating a simple bar chart for two data series. Function plt.subplots(). Options.	
43	Using numpy to store, record, and retrieve data. Numpy .argmax() method.	