Data 6 Python Cheat Sheet

This cheat sheet has been modified from the Data 6 Python Reference and includes all of the functions and table methods that you will need for Quiz 1.

Built-In Python Functions

Function	Description	Input	Output
`str(val)`	Converts `val` to a string	A value of any type (**int**, **float**, **NoneType**, etc.)	The value as a **string**
`int(num)`	Converts `num` to an int	A numerical value	The value as an **int**
`float(num)`	Converts `num` to a float	A numerical value	The value as a **float**
`len(arr)`	Returns the length of `arr`	**array** or **list**	**int**: the length of the array or list
`max(arr)`	Returns the maximum value in `arr`	**array** or **list**	The maximum value the array (usually an **int**)
`min(arr)`	Returns the minimum value in `arr`	**array** or **list**	The minimum value the array (usually an **int**)
`sum(arr)`	Returns the sum of the values in `arr`	**array** or **list**	**int** or **float**: the sum of the values in the array
`abs(num)`	Returns the absolute value of `num`	**int** or **float**	**int** or **float**

NumPy Array Functions

Function	Description	Input	Output
`make_array(v1, v2, ...)`	Makes a NumPy array with the inputted values	A sequence of values	An **array** with those values
`np.mean(arr)`	Calculates the average value of `arr`	An **array** of numbers	**float**: The average of the array
`np.sum(arr)`	Returns the sum of the values in `arr`	**array**	**int** or **float**: the sum of the values in the array
`np.arange(stop)` or `np.arange(start, stop)`	Creates an array of sequential numbers starting at `start` and going up to but excluding `stop`	**int** or **float**	**array**

Table Methods

Function	Description	Input	Output
`tbl.column(col)`	Returns the values in a column	**string** or **int**: the column name or index	**array**: the values in that column
`tbl.num_rows`, `tbl.num_columns`	Computes the number of rows or columns in `tbl`	None	**int**: the number of rows or columns in the table
`tbl.select(c1, c2, ...)`	Creates a copy of `tbl` only with the selected columns	**string** or **int**: the column name(s) or index(es) to be included in the table	**Table** with the selected columns
`tbl.sort(column_name)`	Sorts the rows of `tbl` by the values in the `column_name` column. Defaults to ascending order unless the optional argument `descending=True` is included.	1. **string** or **int**: name or index of the column to sort 2. (Optional) `descending=True`	**Table**: copy of the table with the column sorted
`tbl.where(column, predicate)`	Creates a copy of `tbl` containing only the rows where the value of `column` matches the `predicate`.	1. **string** or **int**: column name or index 2. the value to match to	**Table**: copy of the table with only the rows that match the predicate
`tbl.take(row_indices)`	Creates a table with only the rows at the given indices.	**int** or **array**: indices of rows to be included in the table	**Table**: copy of the table with only the rows at the given indices

Visualization Functions

Function	Description	Input	Output
`tbl.barh(categories)` or `tbl.barh(categories, values)`	Displays a horizontal bar chart with bars for each category in the column `categories`. `values` specifies the column corresponding to the size of each bar, but is unnecessary if the table only has two columns	1. **string**: name of the column with categories 2. (Optional) **string**: name of the column with numerical values	None: draws a bar chart
`tbl.hist(column)`	Generates a histogram of the numerical values in `column`.	**string**: name of the column	None: draws a histogram
`tbl.plot(x_column, y_column)` or `tbl.plot(x_column)`	Draws a line plot consisting of one point for each row in `tbl`. If only `x_column` is specified, `plot` will plot the rest of the columns on the y-axis with different colored lines.	1. **string**: x-axis column name 2. **string**: y-axis column name	None: draws a line graph
`tbl.scatter(x_column, y_column)`	Draws a scatter plot consisting of one point for each row in `tbl`. The optional argument `fit_line=True` can be included to draw a line of best fit through the scatter plot.	1. **string**: x-axis column name 2. **string**: y-axis column name	None: draws a scatter plot