# VerticaPy Lesson Series

## Introduction

This document is an overview of how to use the Jupyter Notebook templates to create lessons on data science concepts and VerticaPy. The core page types are as follows:

* **Course**: Contains video content and covers prerequisites and lesson goals.
* **Module Outline**: Contains a list of the lessons (including exercises, if any).
* **Lesson**: A lesson page goes into detail about a topic, or a portion of a topic. Complex topics often have several lesson pages; for example, you could break up a topic into a theory, application, and exercise page with sample problems.

A set of example pages (theory, application, exercise) is provided in the **/Data Science Essentials/LinearRegression**.

## Prerequisites

* ipywidgets (7.6.5 preferred)
* Ipython
* voila (0.3.6 preferred).

**Note**: The current version of vertica-demo does not meet these prerequisites, so you should consider creating your own environment. Making your own environment entails running VerticaPy and the provided code, and copy-pasting code and output into the Jupyter Notebook.

Basic usage

* To view an existing Jupyter Notebook's webpage layout, open the Notebook and click the **voila** button on the top ribbon. If links do not work, their references need to be updated.
* To edit Jupyter Notebook cells, double-click them. To create new cells, copy the type of cell and paste them; this helps preserve styling between cells.

## Page Types

The following sections go into detail about the core page types and how they should be structured.

### Course

**Video**: Each course page should have a video that demonstrates the course topic(s). The video file must be stored in the ***course*\_title/Figures** directory and named **Video\_1.mp4**.

**Description**: Add a brief, high-level description of the course topics, focusing on the motivation and its applications.

**Highlights**: A short list of the highlights of the course.

**Difficulty and Time**:You can add time in minutes or hours. Difficulty levels are: Easy, Intermediate, Hard.

**Prerequisites**: List prerequisites, if any. Use bullet points if there is more than one.

**Goals**: List the goals of the course, focusing on what a user might want to get out of completing the course.

**Modules:** A list of the lessons as hyperlinks. To get the hyperlinks for each page, use Voilà and copy the address of the page. For example, to get a hyper link of **Module\_v1\_2.ipynb**,copy and replace the existing address with the URL: <http://localhost:8888/voila/render/Documents/Template/Enablement%20Template/Module_v1_2.ipynb>.

Note that this web address contains "voila" after the host and port.

### Module Outline

Tasks:

* Update the lesson title and difficulty level.
* Add a video highlighting important aspects of all the lessons in the module. The instructions of how to create the video will be shared separately.
* Update the hyperlinks for the lessons and exercises.

### Lesson

Lesson pages contain the detailed content and are the primary focus of a course. A course can have several lesson pages.

Tasks:

* Update the **Lesson Name** and **Sub Module Name**.
* Update the hyperlink for the “Go Back to Main Page” so it points to the course page
* Update the estimated time to complete the lesson.
* Add a brief introduction and explain why the lesson is important.
* Update the **Table of Contents** and add references to each header, where each header contains: **<a id="*CELL\_NAME*"></a>** and each link in the **Table of Contents** contains: **text (#CELL\_NAME)**.
* Explain the lesson goals.
* If you use images, add them to their own separate cells so they can be referenced individually.
* To add interactive, multiple-choice "Knowledge Checks," use the **create\_multipleChoice\_widget()** function
* To add code snippets in Markdown, use the following format in a cell:

'''Python

***python code***

'''

* To add a video, reference the video path.
* At the end of the lesson, add the author’s name and contact information.
* Add any citations used.

#### Exercise

An exercise is a special type of lesson page that gives the reader some problems to solve. The types of questions available are:

* Multiple-choice
* Short numeric answer

## Page Names

Pages should be named according to the following format:

***CourseName*\_*ModuleName*\_*LessonName***

For example:

Essentials\_LinearRegression\_Theory

In addition, each module should have its own directory inside the **Course** directory.

## Directories

All Jupyter Notebook pages should be placed inside their respective **Module** directories, which itself is located in the **Course** directory. That is, the hierarchy should be:

**Course folder** > **Module Folder** > **Module-related pages**

For example, the linear regression module contains three pages, and these pages are inside the **Data Science Essentials** directory.

Modules can contain additional directories **Figure** and **Data** directories for for images/icons and data, respectively.

## Courses and Curriculums

Data science is a vast field, so the goal is to focus on the most common topics for now. Anyone is welcome to suggest a course or curriculum.

Data Science is a vast field with lots of rabbit holes, so we want to stick to the most common ones initially. Anyone is welcome to suggest a course or curriculum.

For example, the curriculum for the **Data Science Essentials** course is as follows (each bullet is a module and sub-bullet is a lesson):

**Data Science Essentials**

* *Overview of Data Science*
  + Basic terminology
  + Datasets
  + vDataFrame
* *Basic data preparation*
  + Data formats (csv, image, text, etc.)
  + Basic operations
    - Impute
    - Null or missing values
    - Normali ze
    - Concatenate and Transform
  + Advanced operations
    - Outlier detection
  + Test/Train split
* *Basic data exploration*
  + Visualizations
    - Types of plots
  + Dimension reduction (TSNE, PCA)
* *Advanced statistics*
  + Hypothesis Testing
  + Bootstrapping
  + Bayes Rule
* *Linear Regression*
  + Theory
  + Example
  + Exercise
* *Classification*
  + Theory
  + Example
  + Exercise
* *Project*