Creating Databases and Tables with Hue

In Hive and Impala, each table belongs to a specific database. Databases are helpful for organizing tables; by using multiple databases, you can have tables with the same name in different databases. In this way, they serve as *namespaces*. Even if you don't use tables with the same name, separating databases is helpful for organization and for restricting user access to data they don't need to access, or should not access.

Creating databases and tables using Hue's Table Browser is fairly easy. As you go through this reading, or after you've read through it once, use the VM to create some test databases and tables. You can drop (delete) the test databases and tables when you're done.

To perform the steps described in this reading, you will need to use Hue on the VM. If you do not already have the VM installed and running, please follow the instructions in the reading *Downloading and Installing the VM* in the first week of this course. Then open the web browser on the VM and click the link for Hue in the bookmarks toolbar.

After this reading, the remainder of this week will be about using SQL commands to create databases and tables. After you've learned both methods, you can choose which method you prefer for a given task.

Creating a New Database

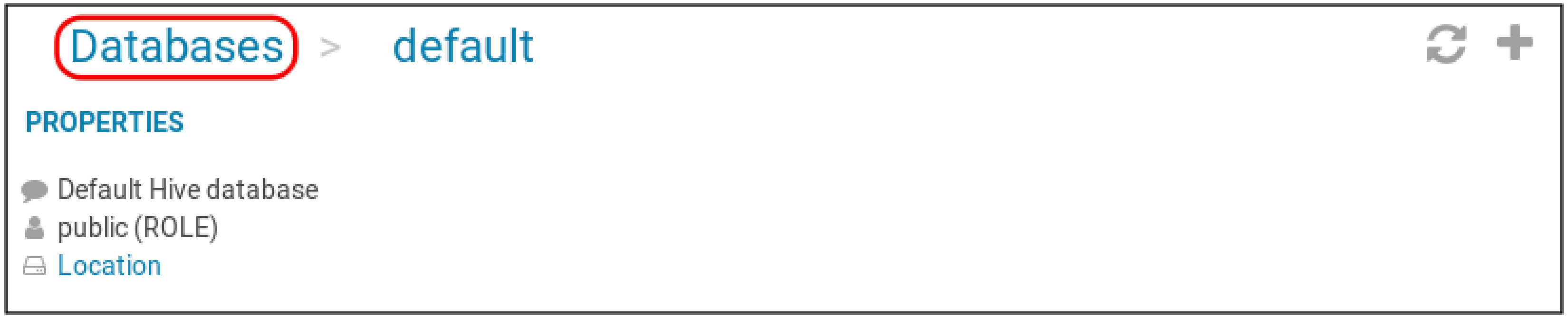
You can start the process for creating a database by going directly into the Table Browser, or by using the data source panel. Try both, and see which you prefer.

When you create a database using these methods, Hive or Impala creates the directory /user/hive/warehouse/*databasename*.db (where *databasename* is the name you entered) unless a different location is specified. Tables created in a database will be placed as a subdirectory within this database directory.

Using the Table Browser

Enter the Table Browser by clicking the hamburger menu (three horizontal lines) and choosing Browsers > Tables. The main (center) panel will show the default database for Hive and Impala. To add a new database:

1. Click Databases in the breadcrumbs at the top. (See Figure 1 below.)
2. Hover over the + symbol on the far right; it should say Create a new database. Click that symbol.
3. Give your database a name such as test, and (optionally) a description in the appropriate field.
4. *Optional:* If you want to store the database files in a location other than HDFS (in S3, for example), uncheck the Default Location box and supply the location.
5. Click Submit.
6. The Task History pop-up will appear; the top should say Creating database *name* with a green line underneath. (See Figure 2 below.) Click the x on the right to dismiss the window.
7. Verify the creation was successful by clicking Databases in the breadcrumbs, and note that your new database now appears in the list. If you like, you can check the HDFS file structure to see that /user/hive/warehouse/*name*.db exists.
8. To drop (delete) your test database, check the box next to it and click the Drop button above the list, then click Yes. This will drop the database and delete the directory from HDFS. Be careful not to drop any databases that have tables in them!



Default Hive database
public (ROLE)

Figure 1

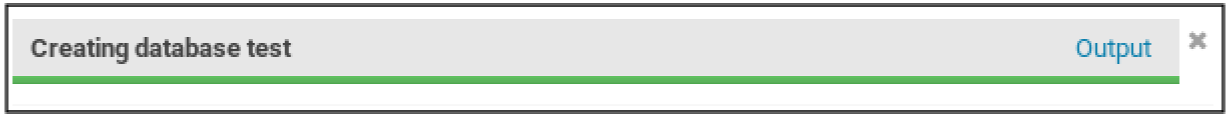

Creating database test

Figure 2

Using the Data Source Panel

The data source panel appears with all areas of Hue, so you can easily use this method without the need to switch to the Table Browser first.

1. Be sure the data source panel is in the database mode (and not in the files mode). The database icon (stacked disks) should be blue. (See Figure 3 below.)
2. Navigate to the Impala source. (See Figure 4 below. You can also use Hive, and the database will be available for both engines.)
3. Hover over the + symbol; it should say Create database. Click that symbol. *(Note: From this point on, the steps are the same as in the previous section.)*
4. In the main panel, give your database a name such as test, and (optionally) a description in the appropriate fields.
5. *Optional:* If you want to store the database files in a location other than HDFS (in S3, for example), uncheck the Default Location box and supply the location.
6. Click Submit. As in the previous section, the Task History pop-up will appear. Click the x to dismiss it.
7. You'll now be in the Table Browser, but in the default database. Verify the creation was successful by clicking Databases in the breadcrumbs, and note that your new database now appears in the list. *You can also navigate in the data source panel back to the Impala source and see the list of databases there.*
8. To drop your test database, check the box next to it and click the Drop button above the list, then click Yes.

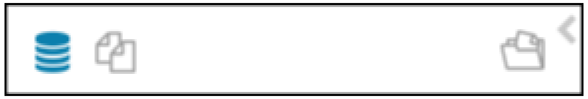
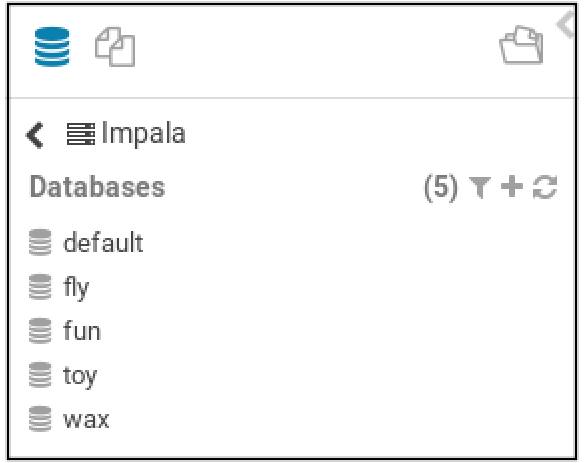


Figure 3



Impala

Databases (5)

default
fly
fun
toy
wax

Figure 4

Creating a New, Empty Table

When you create a table, Hive or Impala creates the directory /user/hive/warehouse/*tablename* or, if it's in a database other than default,/user/hive/warehouse/*databasename*.db/*tablename*. This directory will be empty at first; you will learn in a later week of the course how to populate the table with data by loading data files into it.

You will learn how to drop these tables in a later lesson in this Week, so don't worry about dropping the examples you create now.

As with the database creation, you can start this process with the data source panel (Option A, below), or you can go directly to the Table Browser (Option B). Decide which you want to try from your experiences creating test databases:

- *Option A:* On the data source panel on the left, navigate to the database where you want to put the table. In this case, choose the default database in the Impala source. Hover over the + symbol; it should say Create table. (See Figure 5.) Click that symbol.

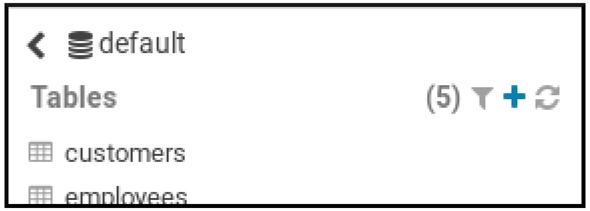


Figure 5

- *Option B:* Enter the Table Browser by clicking the hamburger menu (three horizontal lines) and choosing Browsers > Tables. The main (center) panel will show the default database. (You can switch to a different database if you like, but this isn't necessary for testing purposes. Stay in the default database.) Again, hover over the +symbol on the far right; it should say Create a new table. Click that symbol.

In the main panel, you have the opportunity to indicate a file in HDFS that will be the data for this table. This can be very handy, because Hue will try to guess the columns, including their data types, based on this file. You will see this later in this course, so for now, you'll create an empty table with no data.

1. Click the Type field and change it to Manually; then click the Next button.
2. In the DESTINATION area, name your table. For this example, use default.test, or just test if you are in the default database.

3. For PROPERTIES, you can set the file format and storage location, if desired. For now, leave the defaults (Text format and Store in Default location checked).
4. Under FIELDS, you can specify the columns for your table; click +Add Field for each column and specify the name and data type. Give your test table a couple of columns, such as id and title. You can leave the field type as string for each. You'll learn more about choosing data types later in this course.
5. Click Submit. The Task History pop-up will appear; the top should say Creating table *name* (for the test case, *name* is default.test) with a green line underneath. Click the x on the right to dismiss the window.
6. Verify the creation was successful by looking at the data source panel. Navigate to the database (Impala default) if necessary, and note that your new table now appears in the list. You might need to click the refresh button (two curved arrows) to refresh the display; choose Clear cache if you do this.
7. For your test, check the HDFS file structure to see that /user/hive/warehouse/test/ exists. This directory will be empty, because the table has no data in it.

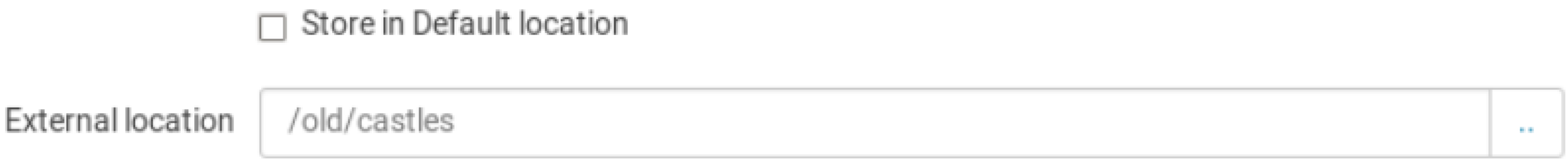
Creating a New Table to Query Existing Data

The steps above described how to create an *empty*table, with no data in it. You can also use Hue to create a table to query data that already exists in HDFS.

For example, in the HDFS directory /old/castles/, there is a file named castles.csv. Review this file: The fields are separated by commas, the names of the columns (name and country) are provided in the header line, and both columns contain character string data. Using the steps below, you can create a table to query this data.

1. Using the data source panel (in the database mode) or the table browser, click the + symbol to create a new table.
2. Click the Type field and change it to Manually; then click the Next button.
3. In the DESTINATION area, name your table. For your example table, use default.castles.
4. For PROPERTIES, you can set the file format. The example file is a text file, so leave the default format (Text) checked.

5. Uncheck the Store in Default location checkbox. When you do this, you will see an External location field appear below it.
6. Enter the HDFS directory path (/old/castles for the example) into the External location field. Alternatively, you can click the .. icon on the right side and use the dialog to select this folder. (See Figure 6 below.)
7. Directly below there, click the Extras icon, which looks like three sliders. You can add an optional description of the data location in the Description field (leave it blank for now). You can also specify the character that separates the fields here by checking the Custom char delimiters checkbox. Check the box and notice that three drop-down menus appear. You can specify different kinds of delimiters here. For this example, you only need the Field drop-down menu. The field separator is already set to Comma (,)which is the field separator (delimiter) used in the file castles.csv, so keep this selection.
8. Under FIELDS, specify the columns for this table; click +Add Field for each column and specify the name and data type. Recall that the file castles.csv, which contains the data for the example table, has two columns: name and country. Add both (in that order), both as string types.
9. Click Submit. The Task History pop-up will appear; the top should say Creating table *dbname.tablename* with a green line underneath. Click the x on the right to dismiss the window.
10. Verify the creation was successful by looking at the data source panel. Navigate to the database (Impala default) if necessary, and note that the new table (castles) now appears in the list. You might need to click the refresh button (two curved arrows) to refresh the display; choose Clear cache if you do this.
11. Hover the cursor over the table name then click the i icon to the right of the table name. Click the Sample tab to verify that the data appears in the sample results.



□ Store in Default location

External location: /old/castles

Figure 6

Limitations

As described above, Hue provides several options for creating databases and tables. However, these options have some limitations. For example, when creating a new table to query existing data stored in text files (as demonstrated above), Hue assumes that the data files have header rows. There is currently no way to specify in Hue that the data files do *not* have header rows.

To overcome these and other limitations of Hue, you can instead use SQL commands to create databases and tables. Furthermore, using SQL commands (instead of Hue user interface actions) is a more systematic way of performing tasks like this. SQL commands can be scripted, automated, and scheduled. By saving your SQL commands in a file, you can effectively document the steps you performed, and you can make the steps reproducible.

In the next reading, you’ll learn how to create databases and tables by running SQL statements.