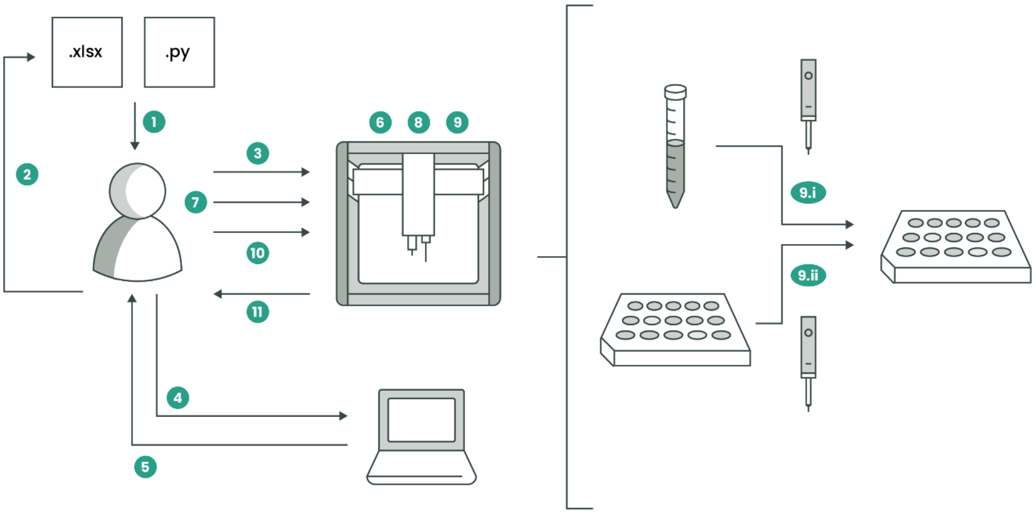
Instructions to use



1. Download the Excel template and Python script.
2. Fill the Excel file with the values of your variable to customise the experiment according to the guidelines at the [end of this document](#).
3. Send the file to the robot that you are going to perform the protocol. If you are using a Unix system, you can send the file using the command `scp` to the folder `/data/user_storage`

   The command should look similar to

   ```
   scp -i [password file] VariablesCounterSelection.xlsx
   root@[IP_Robot]:/data/user_storage
   ```

   For more information about sending files to an Opentrons robot visit https://support.opentrons.com/s/article/Copying-files-to-and-from-your-OT-2-with-SCP
4. Import the Python script to the Opentrons App. A warning will appear on the protocol simulation because the script is not designed to read the Excel variable file from your computer but from the robot system.
5. Start setup in the robot that has your customised Excel file.

   The Opentrons app will simulate the script in the chosen robot, providing you with an error message if something is wrong or, in case the run can be done, with the required labware layout and needed reagents. The colours of the reagents can change due to them being generated randomly, but the volumes will always be the same given the same Excel file.

The volumes of the reagents are exactly what the run needs, so it is recommended to pour in the 15mL falcon tube between 1 and 2 mL more of that reagent.

On the other hand, the liquid of the samples in the source plates corresponds to 90% of the max volume that the well can hold, it is only a recommendation, it does not mean that that sample is going to be selected or that the showed volume is the needed for transferring to the final plates. Nevertheless, is a good practice to have more volume in the wells that is needed due to the pipetting error that the chosen pipettes could have, and the fact that the OT does not have a sensor to check if the volume is picked or dispensed.

6. Set the different reagents and labware as the App shows. Both reagents and different labware will have labels to help you recognise where every one of them should be placed.
7. Perform a labware offset, ensuring all the labware is calibrated correctly.
8. (Optional) Turn on the HEPA filter to lower the chances of contamination.
9. Close the door of the Opentrons and press the button 'Start run'. The Opentrons robot will follow the following stages:
    i) Transfer media to the final labware(s) with a single channel pipette in case is established like that in the variable file.
    ii) Transfer the selected samples, the ones which fulfil the stated criteria, to the final labware (s) with a single channel pipette.
10. Wait for the protocol to finish and, if needed, change the tip racks as stated in the App.
11. Import the maps of the selected colonies in the final labware the same way that you transferred the variable file to the Opentrons system. It should look like the following command:

```
scp -i [password file]
root@[IP_Robot]:/data/user_storage/[name_map].csv [user
computer path]
```

The name of the file is settled in the Excel variable file, in the sheet 'PerPlateVariables' with an CSV file extension. You will have 1 map for every source plate.

Guidelines to fill in the Excel file

General Variables Sheet

API Name Source Plate

- **Type of Value**: String
- **Description**: Opentrons API name of the labware with the samples that can get selected.
  This labware source should have the exact dimensions as the sheets that will correspond to the values that are going to be compared to the threshold value.
- **Example**: `vwrblueprintdepth105_96_wellplate_390ul`

API Name Final Plate

- **Type of Value**: String
- **Description**: Opentrons API name of the final labware in which the selected samples of the source plates will be transferred in addition to the set reagents. Make sure that the selected samples can fit in this labware.
- **Example**: `vwrblueprintdepth105_96_wellplate_390ul`

API Name Rack 15mL Falcon Reactives

- **Type of Value**: String
- **Description**: Opentrons API Name that will contain the 15mL flacon tubes that will have the different media.
  If there are no reactive in "Name Reactives" this cell can be left empty.
- **Example**: `opentrons_15_tuberack_falcon_15ml_conical`

Name Reactives

- **Type of Value**: String | List
- **Description**: Name of all the types of media separated by commas. Don't put spaces or special characters in the names, and make sure that the names correspond to the same ones used in the other sheets of the file.

If this variable is left empty both variables ‘Volume per Reactive (uL)’ and ‘Reactives Per Plate’ will be assumed as empty and not taken in account.

- **Example**: `Glycerol, Water`

Number of Source Plates

- **Type of Value**: Number (Integer)
- **Description**: The total number of source plates that will contain samples.
- **Example**: `2`

Volume per Reactive (uL)

- **Type of Value**: Number | List
- **Description**: Volume (s), in uL, that will correspond to the volumes that are going to be transferred of each media wrote in “Name Reactives”. Volumes to transfer should following the same order as the reagents in the variable “Name Reactives” and should be separated by commas. This value or values cannot be set as 0, in case no reactive want to be mixed with the selected samples just leave empty the value of ‘Reactives Per Plate’.
- **Example**: `25, 50`

Pipette Variables Sheet

API Name Right Pipette

- **Type of Value**: String
- **Description**: Opentrons API name of the pipette that will be in the right mount of the Opentrons arm.
  If this variable is empty both ‘API Name Tiprack Right Pipette’ and ‘Initial Tip Right Pipette’ will be assumed to be empty.
- **Example**: `p300_single_gen2`

API Name Tiprack Right Pipette

- **Type of Value**: String
- **Description**: Opentrons API name of the tip rack associated with the right mount pipette.

This script does not check that the pipette and the tip rack are compatible. In case of pipette and tip rack incompatibility errors like "Arc out of bounds in the Z-axis" could happen.
If the tip rack from the right and left pipette are the same, both pipettes will pick up from the same tip rack, and the initial tip should be the same.

- **Example**: `opentrons_96_tiprack_300ul`

Initial Tip Right Pipette

- **Type of Value**: String
- **Description**: First tip that will be picked by the right pipette.
- **Example**: `A1`

API Name Left Pipette

- **Type of Value**: String
- **Description**: Opentrons API name of the pipette that will be in the left mount of the Opentrons arm.
  If this variable is empty both 'API Name Tiprack Left Pipette' and 'Initial Tip Left Pipette' will be assumed to be empty.
- **Example**: `p20_single_gen2`

API Name Tiprack Left Pipette

- **Type of Value**: String
- **Description**: Opentrons name of the tip rack from which the left pipette will pick up tips from.
  This script does not check that the pipette and the tip rack are compatible. In case of pipette and tip rack incompatibility errors like "Arc out of bounds in the Z-axis" could happen.
  If the tip rack from the right and left pipette are the same, both pipettes will pick up from the same tip rack, and the initial tip should be the same.
- **Example**: `opentrons_96_tiprack_20ul`

Initial Tip Left Pipette

- **Type of Value**: String
- **Description**: First tip that will be picked by the left pipette.
- **Example**: `B2`

Replace Tiprack

- **Type of Value**: Boolean
- **Description**: Value that will determine that, in case of more than one tip rack needed for the run of the protocol, the tip rack will be replaced and not added to the labware layout.
  If the right and left mount pipettes tip racks are the same, and this variable is set to True, only one slot in the layout will be occupied by tips.
  This variable only accepts two values, True (tip racks will be replaced) or False (tip racks will be added to the layout).
- **Example**: `True`

Per Plate Variables Sheet

Threshold Selection Value

- **Type of Value**: Number
- **Description**: Number that will act as a criterion to select the samples, one of the values should be lower and the other higher for a sample to be selected and transferred to the final labware.
- **Example**: `0.2`

Name Sheet Selection Value<Threshold

- **Type of Value**: String
- **Description**: Sheet name in the Excel file that will contain the values of the source plate samples that, to be selected, need to be equal or lower than the number stated in the variable “Threshold Selection Value”.
- **Example**: `ValuesMinorThreshold_SourcePlate1`

Name Sheet Selection Value>Threshold

- **Type of Value**: String
- **Description**: Sheet name in the Excel file that will contain the values of the source plate samples that, to be selected, need to be equal or greater than the number stated in the variable “Threshold Selection Value”.
- **Example**: `ValuesMajorThreshold_SourcePlate1`

Reactives Per Plate

- **Type of Value**: String | List
- **Description**: Name of the reagent or list of reagents separated by commas that the selected samples will be mixed within the final labware.
  If we want to create two final plates with the same reagent, we should put it twice in that list.
  If this value is left empty the program will create a plate only with the "Volume Transfer Sample (uL)" without any additional reagent.
- **Example**: `Glycerol, Water, Water`

Well Start Final Plate

- **Type of Value**: String
- **Description**: First well in the final labware that will be filled with the combination reactive + selected sample in the final labware. This well will be the first one to get filled in every final labware created from this source plate.
- **Example**: `A2`

Final Map Name

- **Type of Value**: String
- **Description**: Name of the map that will be exported by the protocol that will contain the positions of the selected sample in the final labware.
  All final labware will have the same layout of selected samples for a source plate, that is why only 1 map is provided per source plate.
  This name should not include the extension because will always be a CSV file.
  The output file will be stored in the path or directory /data/user_storage of the robot where the protocol has been run.
- **Example**: `map_selectedSamples1`

Volume Transfer Sample (uL)

- **Type of Value**: Number
- **Description**: Volume, in uL, that will be transferred from the colony samples to the final labware.

Make sure that there is enough volume in each well of the source plate to be transferred to all stated final plates and an extra volume to make sure the pipette takes the required volume because the protocol won't check that requirement.

- **Example**: `50`

* If needed, this sheet can have more columns to the right, this meaning that if the Opentrons have more than 11 available slots, the sheet can be expanded and filled. This column adding does not guarantee that more final plates can be created because that result depends on more variables, such as the tip rack replacement value.
There is a need of as much 'Name Sheet Selection Value > Threshold' and 'Name Sheet Selection Value < Threshold' sheets as 'Number of Source Plates'.
The columns of the plate can have any name, these names will be included in the final layout that you will be able to see in the OT-App to help you recognize where each source plate has to be placed. Nevertheless, if there is an error that is cause by any of the variables of this sheet, the named of the source plate will be given by the index of the column, being Source Plate 1 the one in furthest on the left.

Selection Value Sheet(s)

Sheet with the name set in the variable "Name Sheet Selection Value > Threshold" and "Name Sheet Selection Value < Threshold" in the Sheet "PerPlateVariables".

This sheet should have the same dimensions (rows and columns) as the labware set in the variable "API Name Source Plate" in the sheet "GeneralVariables". For instance, a labware that has 8 rows and 12 columns, should have 2 sheets with those dimensions.

The sheet should not have the names of the columns, only the values that are going to be compared.

No cell of these sheet should be left empty and only numbers are accepted as suitable values for the protocol to run correctly.

For an example of how the sheet should look like check the attached Excel file example