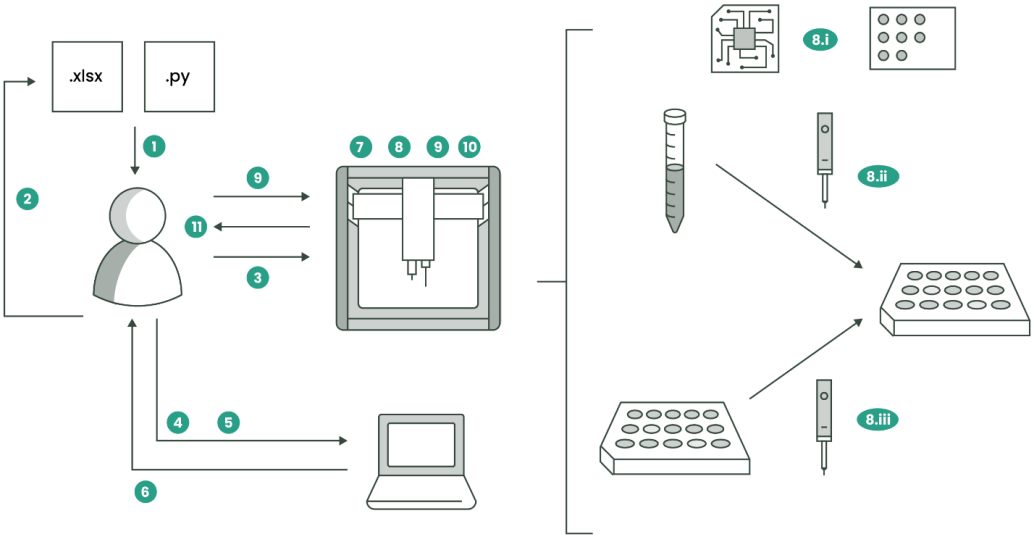
# Instructions to use




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] VariablesMergeSamples.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.
6. The Opentrons app will simulate the script in the chosen robot, providing you with an error message if something is wrong or with the required labware layout and needed reagents.

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.

7. Set the different reagents and labware as the App shows. They will have labels to help you recognise where every reagent and labware should be placed.
8. Perform a labware offset, ensuring all the labware is calibrated correctly.

   Close the door of the Opentrons and press the button Start. The Opentrons robot will follow the following stages:

   i) Opentrons will decide which samples will transfer considering the variables set by the user.

   ii) Transferring media (Optional)

   iii) Transfer the selected samples from the source plate (s) to the final plate (s).

9. If needed, change the tip racks as stated in the App.
10. Wait until the protocol has finished.
11. Import the map of the selected colonies in the final labware the same way that you transferred the variable file to the Opentrons system. It should look similar to the following command:

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

The name of the file is set in the sheet 'GeneralVariables' of the variables file and will always have an EXCEL file extension. There will be one sheet in the final document for each final plate stating the slot where they were.

# 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(s) containing the samples to be selected. The labware should have the same dimensions as the "Name Sheet Map Identifiers" sheet(s), in other worlds, both should have the same number of columns and rows.
- **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 and the reagent can be distributed.
- **Example**: `armadillo_96_wellplate_200ul_pcr_full_skirt`

### 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 reagent tube(s) needed for distributing.
  In case the variable "Volume Reactive Transfer (uL)" is left empty or as a 0, this cell can be left empty.
- **Example**: `opentrons_15_tuberack_falcon_15ml_conical`

### Volume Reactive Transfer (uL)

- **Type of Value**: Number
- **Description**: Volume, in uL, that the reagent will dispense in every final well. This value can be 0 or left empty, in which case, it will be assumed 0 and the variable "API Name Rack 15mL Falcon Reactives" will be ignored.
  Make sure that this volume can be picked by at least 1 of the set pipettes, i.e., that the min volume of one of the pipettes is equal or lower to this value.
- **Example**: `75`

### Number of Source Plates

- **Type of Value**: Number (Integer)
- **Description**: The total number of source plates containing samples to transfer
- **Example**: `3`

### Name File Final Map

- **Type of Value**: String
- **Description**: Name of the output file that will be stored in the robot's system directory /data/user_storage.
  The output file will be a XLSX file containing the final plate layout in which each sheet will be the map of one of the final plates to be created.
- **Example**: `map_final_merge`

### Well Start Final Plate

- **Type of Value**: String
- **Description**: Name of the first well where a sample is transferred in the final plate. Make sure that this well exists in the labware set in the variable "API Name Final Plate".
- **Example**: `A3`

## 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 left empty both 'API Name Tiprack Right Pipette' and 'Initial Tip Right Pipette' will be ignored.
- **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 left empty both 'API Name Tiprack Left Pipette' and 'Initial Tip Left Pipette' will be ignored.
- **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**: `A1`

### 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 pipette 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

### Name Sheet Map Identifiers

- **Type of Value**: String
- **Description**: Sheet name that will contain the values of the source plate samples. These names will be the one reflected in the final map file.
  These sheets need to have the same dimensions as the source labware, in other words, need to have as many columns and rows as the labware set in the variable “API Name Source Plate”. Note that the name of rows and columns shouldn’t be specified.
  Wells without sample should be left empty.
  There is a need of as many Map Identifiers Sheets as source plates and should have the same names as in this variable.
- **Example**: `pBLAM126T`

## Type of Sample Selection

- **Type of Value**: first, last, random
- **Description**: Type of selection that will be imposed to the source plate samples.
  The considered wells will be from the "First Well Consider Take" even if the map has a name in the cells before it. Wells are considered top to bottom, left to right.
  The wells that have a hyphen in the maps will not be considered when picking samples because they are considered as empty.
  This variable can only have 3 values:
  - **first**: the first "Number Samples Pick" will be taken from the well "First Well Consider Take".
  - **last**: the last "Number Samples Pick" will be taken from the well "First Well Consider Take" to the final well of the labware. The samples will be taken back to front.
  - **random**: randomly "Number Samples Pick" samples will be selected from the wells with samples as state in the map of the source plate taking account only the wells from "First Well Consider Take".
- **Example**: `first`

## First Well Consider Take

- **Type of Value**: String
- **Description**: First well from the plate that will be considered to pick a sample. All wells before this one will not be considered when selecting wells to take a sample from. Take in account that wells are considered left to right and up to down.
  For instance, if I put as first well the value B2, the wells on the first column and A2 will not be considered, even if they have a sample.
  Take in account that the number of well from this value should be greater or equal to the number set in the variable "Number Samples Pick".
- **Example**: `B1`

## Number Samples Pick

- **Type of Value**: Number (Integer)
- **Description**: Number of samples that will be transferred to the final plate.
- **Example**: `12`

### Volume Transfer Sample (uL)

- **Type of Value**: Number
- **Description**: Volume that will be transferred to the final plates of the selected samples. This value cannot be set as 0.
  Ensure that you can transfer the volume with the established pipettes**.**
- **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 the final plates can be created because that result depends on more variables, such as the tip rack replacement variable.
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 must be placed. Nevertheless, if there is an error that is cause by any of the variables of this sheet, the name of the source plate will be given by the index of the column, being Source Plate 1 the one in furthest on the left.

## Map Identifiers Sheet(s)

Sheet with the name set in the variable “Name Sheet Map Identifiers” 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”.

It should have the names of columns and rows correspondent to the rows and column names of the labware, for example, a 96-well plate will have rows with the letter A-H and columns with the name 1-12.

Wells without any sample should be left empty, they will not be taken in account when selecting the samples.

You can see an example of 4 map Identifiers sheets in the attached Excel attached.

**IMPORTANT:** in the final plates you will only see reflected the names that are stated in these sheets so make sure that there are no samples with the same

name within the sheet nor between any other maps because those samples will become indistinguishable in the final plate(s).