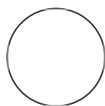# 🌐 OT-2 Counter-Selection V.2

Ana Mariya Anhel[1], Lorea Alejaldre[1], Ángel Goñi-Moreno[1]

[1]Centro de Biotecnología y Genómica de Plantas, Universidad Politécnica de Madrid (UPM)-Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria (INIA/CSIC), Madrid, Spain

Ángel Goñi-Moreno: angel.goni@upm.es

VERSION 2

DEC 22, 2023

biocomp.cbgp

DOI:

Protocol Citation: Ana Mariya Anhel, Lorea Alejaldre, Ángel Goñi-Moreno 2023. OT-2 Counter-Selection. **protocols.io** https://dx.doi.org/10.17504/protocols.io.5qpvor5xdv4o/v2Version created by Ana Mariya Anhel

Protocol status: Working
We use this protocol and it's working

ABSTRACT

This protocol is meant to select the samples from one or more source plate(s) that has 2 different selecting conditions: one condition gives higher values and the other lower values. From these selected samples this protocol create final plates with the samples alone or mixed with different reactive(s).

The instructions for running this protocol are a python script for an Opentrons 2 robot and a excel file with several variables making the protocol modular to reactives, volumes of transfer, type of plates, etc.

In our lab we are selecting samples by their after night OD in 2 different antibiotics and create PCR plates (with water) and 2 stock plates (with glycerol), but it can be used with other selection characteristics such as expression of GFP and with other types of reactives.

This protocol description corresponds to the entry **LAP-ColonyCounterSelection-OT2-1.0.0** of the LAP repository

GUIDELINES

This protocol was developed with python 3.7.1, OT App Software Version 6.3.1 and API level version 2.14 in a Linux 4.14.74 system (these are the OT-2 specifications). In the script several packages are used: pandas (0.25.3), openpyxl (3.1.2), math, random and numpy (1.15.1)

It has been tested with cultures from *Pseudomonas putida KT2440* as part of a counter-selection step in the High-throughput workflow for the genotypic characterization of transposon library variants.
30% glycerol and water have been successfully dispensed using this protocol.

The maximum number of 96-well plates per run given 1 source plates, 1 pipette and replacement of the tiprack set as True is 8 final plates (using 1 tip rack and 1 tube rack for falcons tubes).

MATERIALS

Software

- Python 3.7.1
- opentrons software version 6.3.1
- python packages: pandas (0.25.3), openpyxl (3.1.2), numpy(1.15.1), math, random
- OT-App
- Excel

OT-2 Labware

- Opentrons Tip racks

Equipment	
Opentrons 96 Tip Rack 300 µL	NAME
Tip rack	TYPE
Opentrons	BRAND
-	SKU
https://labware.opentrons.com/opentrons_96_tiprack_300ul?category=tipRack	LINK

Equipment	
Opentrons 96 Filter Tip Rack 20 μL	NAME
Tip rack	TYPE
Opentrons	BRAND
-	SKU
https://labware.opentrons.com/opentrons_96_filtertiprack_20ul?category=tipRack	LINK

- 96-well plates

Equipment	
96-well plates, flat bottom, non treated	NAME
Cell culture plates	TYPE
VWR	BRAND
734-2781	SKU
https://es.vwr.com/store/catalog/product.jsp?catalog_number=734-2781	LINK

- Opentrons Falcon Tube Rack

Opentrons 15 Tube Rack with Falcon 15 mL Conical	NAME
OT Tube Rack	TYPE
Opentrons	BRAND
-	SKU
https://labware.opentrons.com/opentrons_15_tuberack_falcon_15ml_conical/	LINK

- 15mL Falcon tubes

Falcon® Conical Centrifuge Tubes 15mL	NAME
Flaocn Tube	TYPE
Falcon	BRAND
352096	SKU
https://ecatalog.corning.com/life-sciences/b2c/US/en/Liquid-Handling/Tubes,-Liquid-Handling/Centrifuge-Tubes/Falcon%C2%AE-Conical-Centrifuge-Tubes/p/falconConicalTubes	LINK

Equipment

Equipment	
OT-2	NAME
Liquid handler	TYPE
Opentrons	BRAND
OT-2	SKU

Equipment	
HEPA Module	NAME
Opentrons	BRAND
OT-2-HEPA	SKU
https://opentrons.com/modules/hepa-module/	LINK

Equipment	
Single Channel Electronic Pipette (GEN2) 300uL	NAME
Opentrons Pipette	TYPE
Opentrons	BRAND
-	SKU
https://shop.opentrons.com/single-channel-electronic-pipette-p20/	LINK

> ❗ It is important to use HEPA module to work in sterility

BEFORE START INSTRUCTIONS

Note that this protocol needs 2 values to do a selection for each source plate

Files Preparation

1 **Preparing Customized Template**

Preparing the template (a .xlsx) with the specific variables for each experiment.

Here we attach a template of the variable file with several sheets and a PDF file explaining each variable:

1. **GeneralVariables:** variables related mainly to the labware that is going to be used
2. **PipetteVariables:** variables related to the pipettes that are going to be used
3. **PerPlateVariables:** variables related to the specifications of each source plate
4. **Sheet(s) for comparable values (OD, fluorescence, etc):** values that will be compared against the threshold established in the PerPlateVariables sheet -> *Not in the template but should be present in the final file*

▢ Template-VariablesCounterSelection.xlsx `17KB`

▢ CounterSelectionInstructions.pdf `188KB`

1.1 *Fill the template with the corresponding values*

1.2 *Store it with the name VariablesCounterSelection.xlsx*

> **Note**
>
> The file should be spelt **exactly** *VariablesCounterSelection.xlsx* or the Python script won't work correctly

2   Transferring file to Robot

Transfer the *VariablesCounterSelection.xlsx* to the directory */data/user_storage* of the OT robot that we are going to use to perform the protocol.

Note

Before transferring any file to the OT, we need to know the IP of the robot.

This can be obtained in the Networking section of the Device that we will use.
To obtain this info, go in **OT-App -> Devices -> Chosen Robot (three dots) -> Robot Settings -> Networking**

In this tab, you can see 2 types of IP; one is shown if both the robot and you are connected to the same Wifi, and the other is shown if the computer and the robot are connected via USB. Both connections can be used for this step

---

Note

To connect to the robot an **OT-key** should have been previously generated, and it is done with the *ssh-keygen* command and **transferring the public key to the OT.**

For more information about how to generate and set the connection between your computer and the Opentrons robot, visit https://support.opentrons.com/s/article/Setting-up-SSH-access-to-your-OT-2

Here, we present a summary of how to transfer the files in 3 Operative Systems: *Windows, MacOS* and *Linux*

MacOS/Linux
We will use the command line with scp to transfer the file *VariablesCounterSelection.xlsx* to the OT system.

We need to perform the following command

Command

File passing from linux (our computer) to linux (OT raspberry)

scp -i [ot_key] [file] root@[IP_OT]:/data/user_storage

Note

You can face difficulties transferring files in MacOS Ventura (13) and Sonoma (14). These problems can be solved by adding the argument -O (uppercase o) to the command

Command

**Transferring files to OT (MacOS 13 and 14)**

scp -Oi [ot_keypath] [file path] root@[OP_robot]:/data/user_storage

Windows
There are several ways to send files from a Windows to a Linux (for example, with a virtual machine or Windows Powershell in the latest versions of Windows).

Here, we will use **FileZilla** (https://filezilla-project.org/download.php?type=client).

Go to **File -> Site Manager -> New Site -> Change Protocol to SFTP**. Then, introduce in Host the OT IP, change the *Logon Type* to key file, change *User* to root and give the directory where the OT key is. It should look something like this
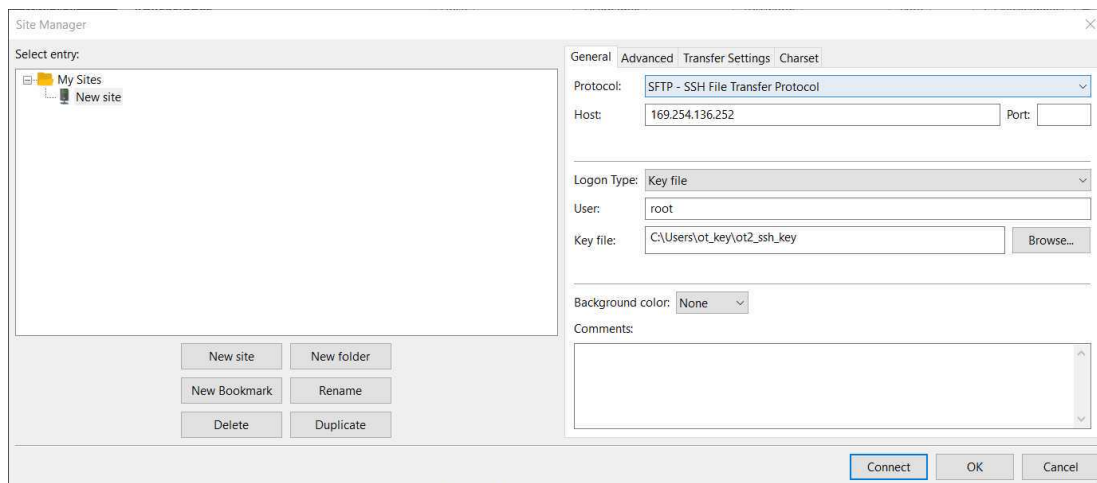
Example of setting the FileZilla to transfer files from Windows (our computer) to Linux (OT)

Then press *Connect*, and we will have a connection between our computer and the robot.

After this connection, we should be able to move the file *VariablesCounterSelection.xlsx* (in our computer) to the directory */data/user_storage* in the robot.

This method can be used as well in some Linux and MacOS

> **Note**
>
> Take into account that the IP of the robot could change, so it is possible that it will be necessary to change the host in these connections from time to time.

3   Adding the custom labware

There is only a need to do this step when labware that you are using is not OT official or is not included in the OT app

3.1   *Creation of .json file*

The description file can be obtained by describing the labware dimensions at
https://labware.opentrons.com/create/

3.2   *Uploading file to the OT App*

In the OT app, we need to perform the following route: **Labware -> Import -> Choose File -> Select file we have created in step 3.1**

> **Expected result**
>
> After uploading the labware you should be able to see the new labware in the Labware tab of the OT App, all custom labware can be found more easily in the category *Custom Labware*

3.3   *Transfer labware files to robot*

If you are using the entry **LAP-ColonyCounterSelection-OT2-1.0.0 or LAP-ColonyCounterSelection-OT2-1.1.0 and custom labware**, an additional step is needed, which is transferring a folder with the custom labware.

We need to create for our custom labware a folder with the API name containing the description file (.json) called 1.json and then transfer that folder to the robot's folder */data/labware/v2/custom_definitions/custom_beta* in a similar way as in the Step 2 but with the difference that is a directory that needs to be transferred and not a file.

> **Command**
>
> **Transferring the custom labware to OT (Linux)**
>
> scp -i [ot_key] -r [directory_custom_labware] root@[IP_OT]:/data/labware/v2/custom_definitions/custom_beta

> **Note**
>
> We do not need to execute this part every time the protocol is used, only when that labware is not included in the OT official labware and these directories are not in the robot.

Prepare Robot OS

4   Install needed packages
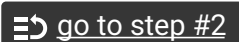
This script needs the package *openpyxl*, which is not installed by default in the OT-2 robots

> **Note**
>
> This step is only needed if the package is not installed in the robot, not in every run of the protocol.
>
> If the package is not installed, an error will appear when running the script in the robot. This error will appear while simulating the script in the app, but you can ignore it.

4.1   *Connect to the robot*

⮑ go to step #2   to find the IP of the robot in which you want to run the script

To connect to the robot, you can do it via ssh with the following command
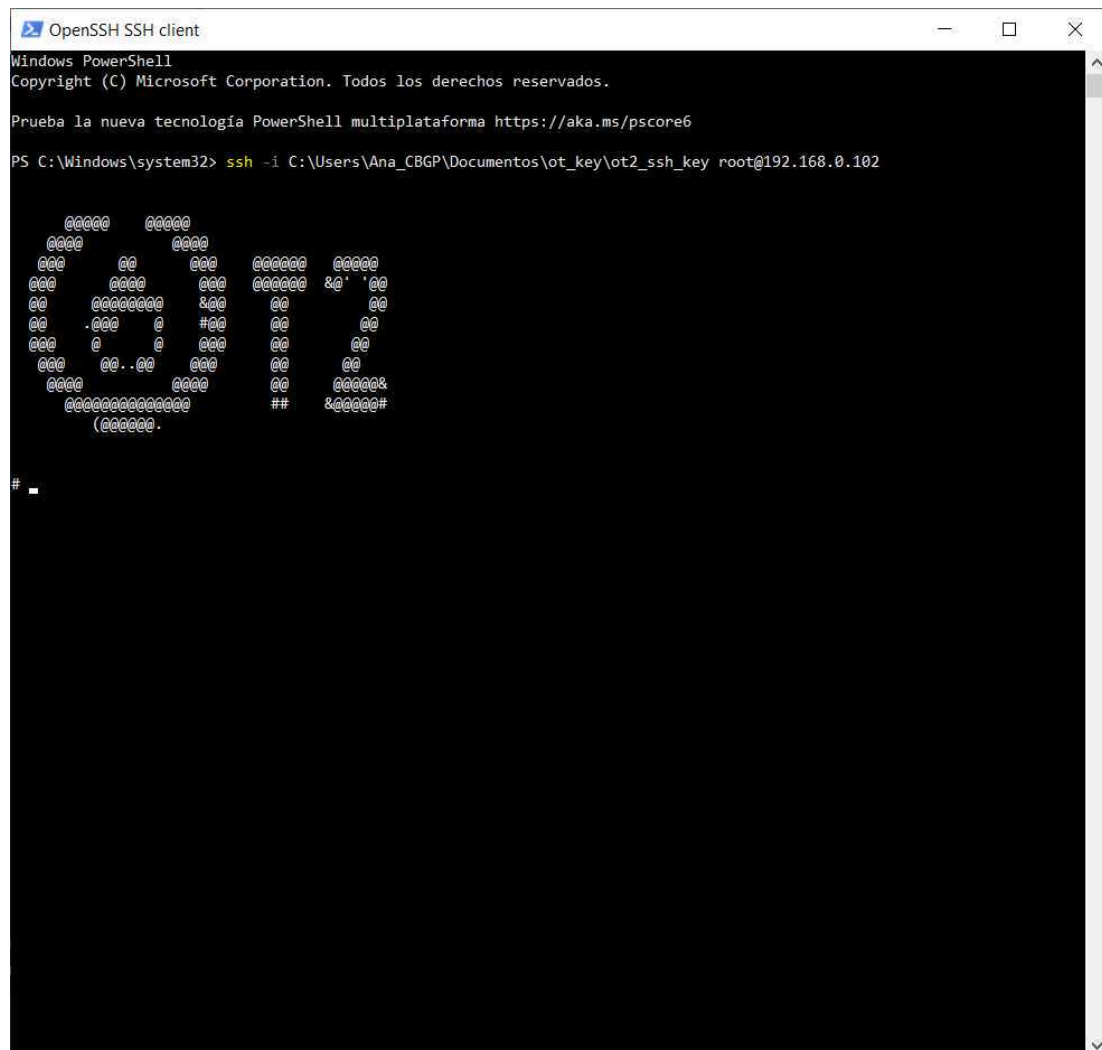
> **Command**
>
> **Connect to Linux based OT via ssh**
>
> ssh -i [path ot_key] root@[Robot_IP]

In Windows, you can do this command in Windows Powershell

If the connection has been successful you should obtain a screen similar to the following image

Drawing obtained when entering an OT-2 system

4.2   *Install the package*

Once inside the robot's system, you need to run the following command

Command

Install openpyxl package (Linux 4.14.74-v7)

pip install openpyxl

Note

For more information about installing packages in the opentrons robots, check the following Opentrons page: https://support.opentrons.com/s/article/Using-Python-packages-in-Python-API-protocols

Running Protocol

5   **Load script in OT-App**

Now that we have transferred the variable files to the robot, we can load the script and run it in the selected robot

Note

This whole step has been developed with version 6.3.1 of the OT-App

Indications may vary from version to version of the opentrons App and the version of the script.

Software	
Opentrons App	NAME
Windows >=10, Mac >=10 , Ubuntu >=12.04	OS
Opentrons	DEVELOPER
https://opentrons.com/ot-app/	SOURCE LINK

5.1   *Load the script in the OT App*

Protocols -> Import -> Drag Python script

> **Note**
>
> The last script version can be found at
> https://github.com/BiocomputationLab/LAPrepository/tree/main/LAPEntries (the name of this
> file is the user's choice). The name of the directory should be **LAP-ColonyCounterSelection-OT2**
> followed by the version.
>
> As well we can find the latest version of the script at
> https://www.laprepo.cbgp.upm.es/repository/ with the same name as in GitHub

Software	
LAP Repository	NAME
https://biocomputationlab.com/	DEVELOPER
www.laprepo.com	SOURCE LINK

The App with version 6.3.1 analyzes your protocol before setting a robot to run, so the labware will not be shown before assigning the protocol to a specific robot when you import it into the App

5.2   *Select Robot to Perform Script*

Click in the protocol -> Start setup -> Choose the OT where the file *VariablesCounterSelection.xlsx* is -> Proceed To Setup

After clicking on Proceed to Setup, you should obtain the labware positions in the *Labware* tab and the reagents, with their corresponding volume, in the *Liquids* tab.

In case the protocol with the set variables cannot run, an error will occur during the run. A lot of errors are contemplated already and have a specific message that gives a hint of what could have gone wrong.

Expected result

A labware setup should look like the following image, where you can find the initial and final plates, the Falcon labware to store the reagents and the corresponding tips
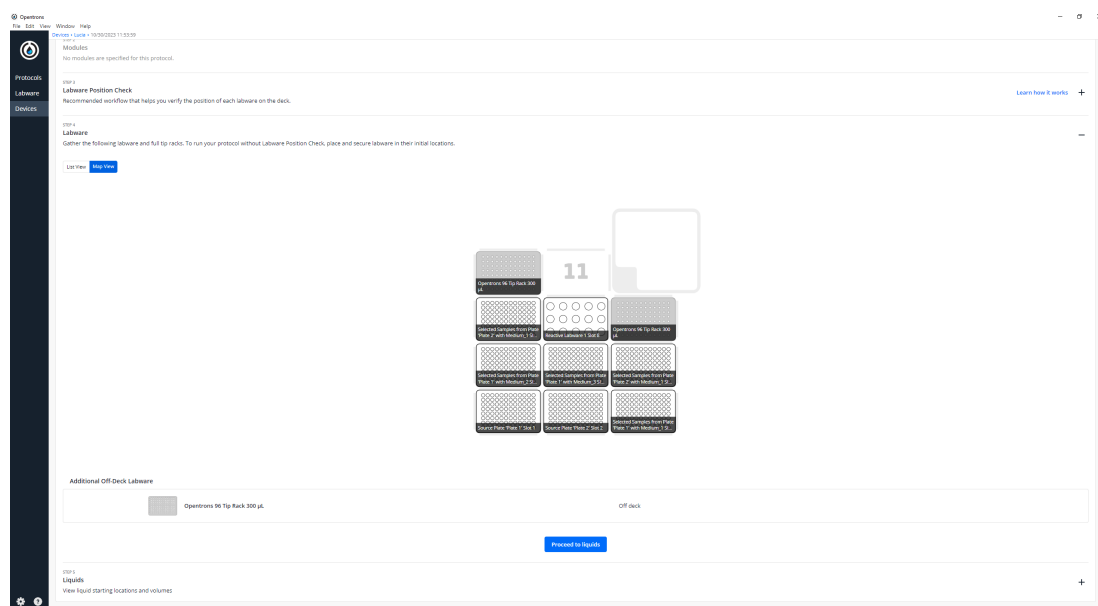
Labware Set-up example of Counter Selection Protocol

Expected result

A liquid setup should look like the following image, where you can find the samples in the initial plates and the reagentsin the Falcon labware
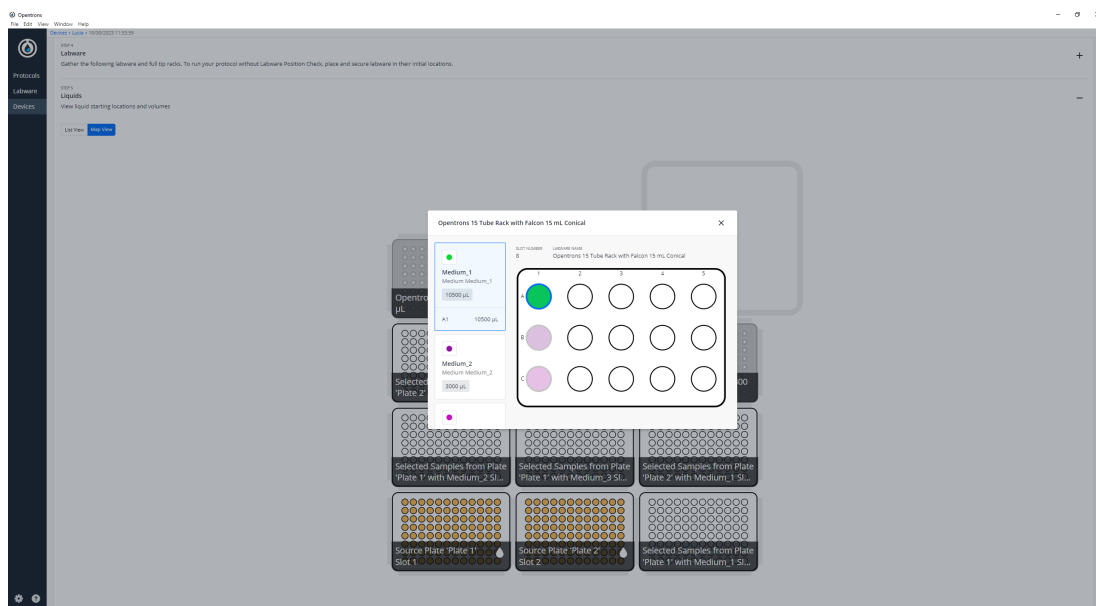
Liquid Set-up Example of Counter Selection Protocol

Note

The volume of the initial samples is established to be 90% of the max volume of the well, but it is only a recommendation, just **make sure that there is enough volume to transfer to all the final plates.**

On the other hand, the volume of the reagents is exactly what is needed, so it is **suggested to pour always more to take into account the error of pipetting**

Note

It is recommended that you perform a labware position check.

You can do it with test plates after loading the script but before cleaning the surface. That way, you reduce the probability of contamination (using the test plates and labware) and pipetting errors (position check).

6   Run Protocol in OT

6.1   *Make sure the needed calibrations are done*

Pipettes, tip racks and tip length calibrations need to be done for the items used in this run

6.2   *Labware position check is performed (if needed)*

✱

6.3   *Clean the surface of the robot with 70% ethanol to clean and disinfect the surfaces*

> **Note**
>
> Check the Opentrons page https://support.opentrons.com/s/article/Cleaning-your-OT-2? for more information about cleaning the OT-2 robot with the proper materials

6.4   *Set the labware and reagents as shown in the OT-App*

6.5   *Start Run*

The procedure that the robot is going to do is mainly divided into 4 parts:

1. Select colonies that comply with the selected parameters (threshold)
2. Distribute reagents to respective plates (if set in the input file)
3. Distribute to all plates the selected samples
4. Generate identity maps (to be exported in thefollowing steps)

After-Running

7   **Retrieve labware from the OT**

8   **Import map(s) from robot**

There will be as many maps as source plates with the following structure of names: [*FinalMapName*].*csv*

To retrieve them, we can ⊐⊃ go to step #2 and reproduce it by transferring the files from the robot to the computer.

All of the maps will be in the directory */data/user_storage*

Example
<div align="right">1h 53m</div>

9   **We want to select the samples that have an OD$_{600nm}$ higher than 0.5 in the Antibiotic Transposition genome file and have an OD$_{600nm}$ lower than 0.5 in the Antibiotic ampicillin plasmid file in one of the source plates and the same selection in the second one but with the value 0.6**

After selecting the samples, we want to transfer those samples to final plates that will have different media

We will use a computer with a Windows 10 system

9.1 Excel template that we can find ⟲ go to step #1 filled and saved with the name
VariablesCounterSelection.xlsx

☐ VariablesCounterSelection.xlsx22KB

9.2 **Upload custom labware to app**

We are using a custom labware called *vwrblueprintdepth105_96_wellplate_390ul* that has been
created with the labware creator that opentrons offers (https://labware.opentrons.com/create/)

☐ vwrblueprintdepth105_96_wellplate_390ul.json11KB
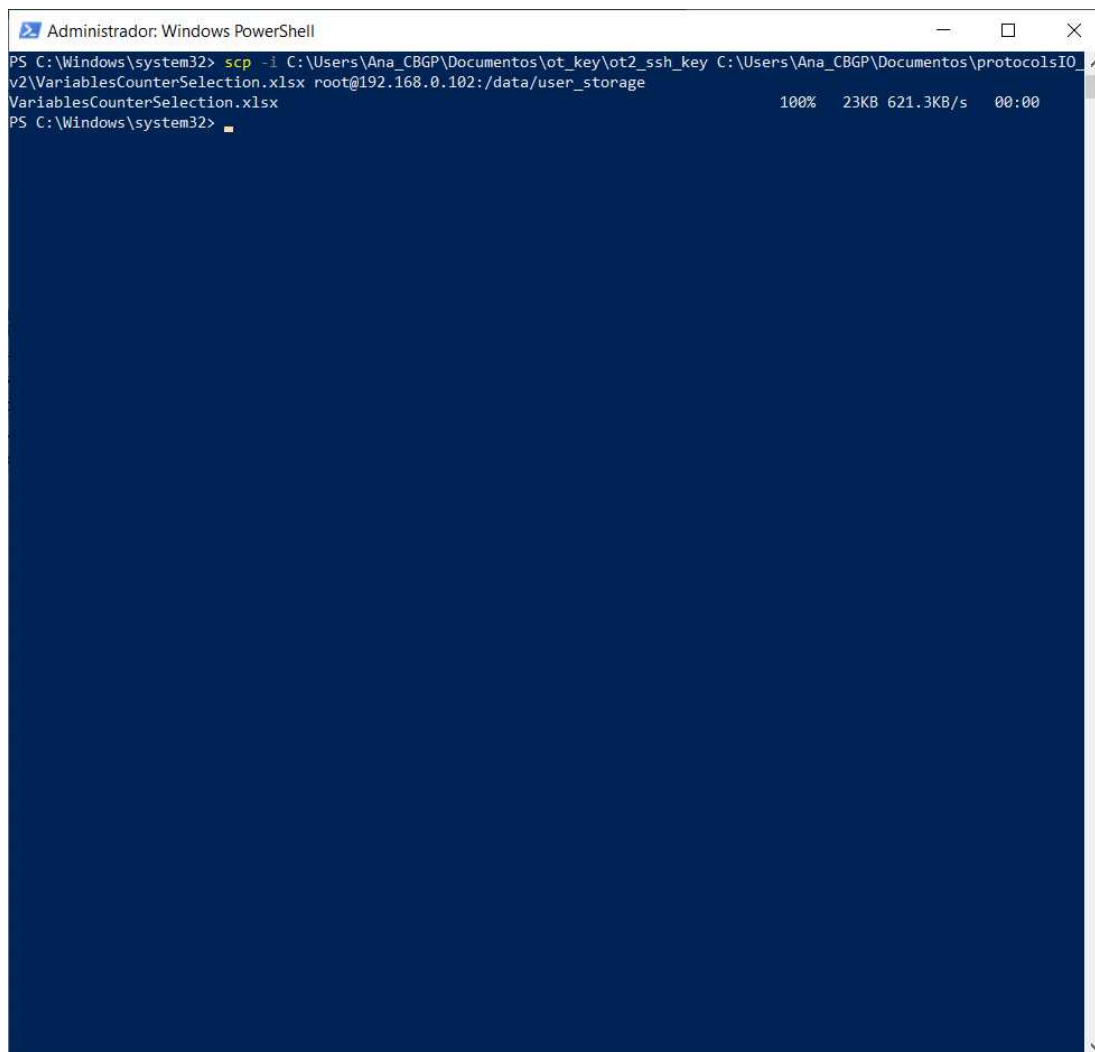
Upload it to the labware and make sure it is loaded in the app
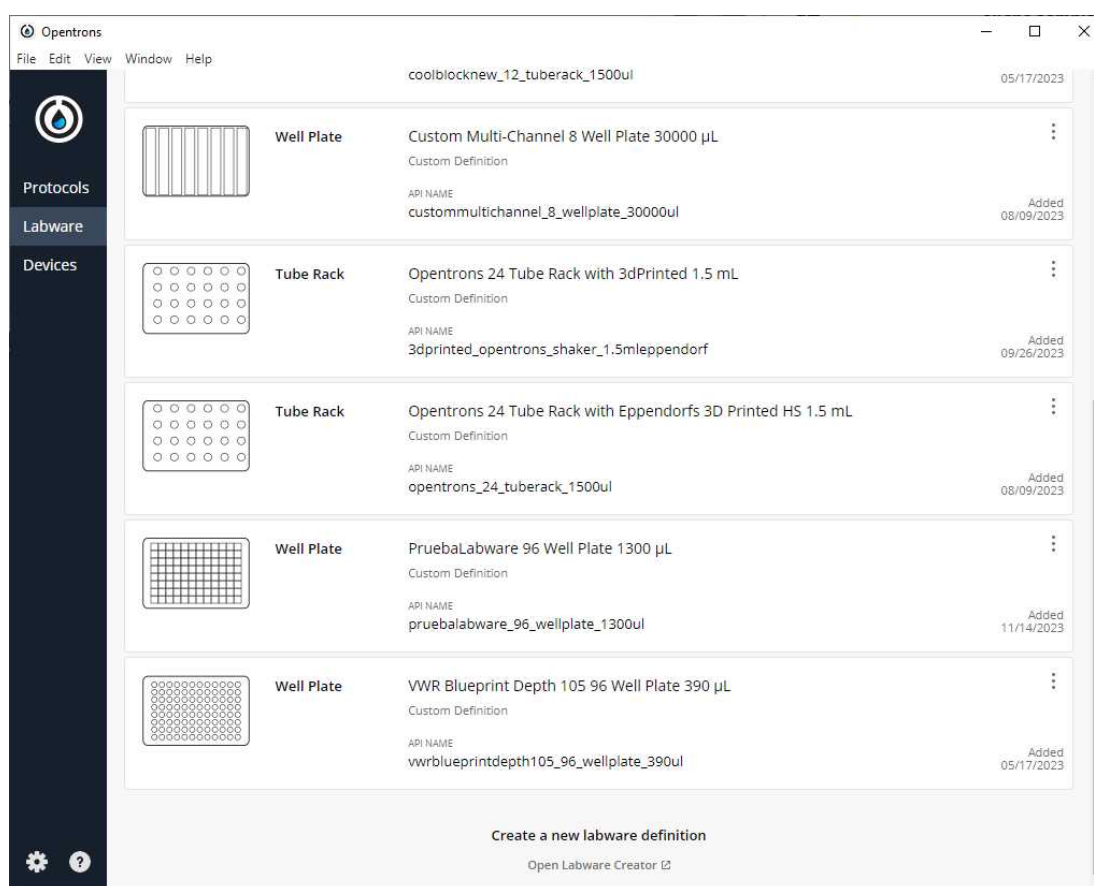
List of custom labware recorded in the Opentrons App

9.3 Export the variable file to *the /data/user_storage* folder in the robot

command line window with scp commands to transfer the variables .xlsx from our computer to the OT-2

9.4 Because we are using version 1.0.0 of the script in this example, we will transfer the directory of the `1m` labware as well (here we have attached a zip, but it is the folder that must be transferred, not the zip)

vwrblueprintdepth105_96_wellplate_390ul.zip 1KB

Command
Transferring the used custom labware to OT (Linux)
scp -i [ot_key] -r vwrblueprintdepth105_96_wellplate_390ul root@[IP_OT]:/data/labware/v2/custom_definitions/custom_beta

9.5 Import the script that we have downloaded from the step ⮐ go to step #5.1 (I named it *ColonyScreening-Example.py*) to the OT-App `30s`

ColonyScreening-Example.py `53KB`
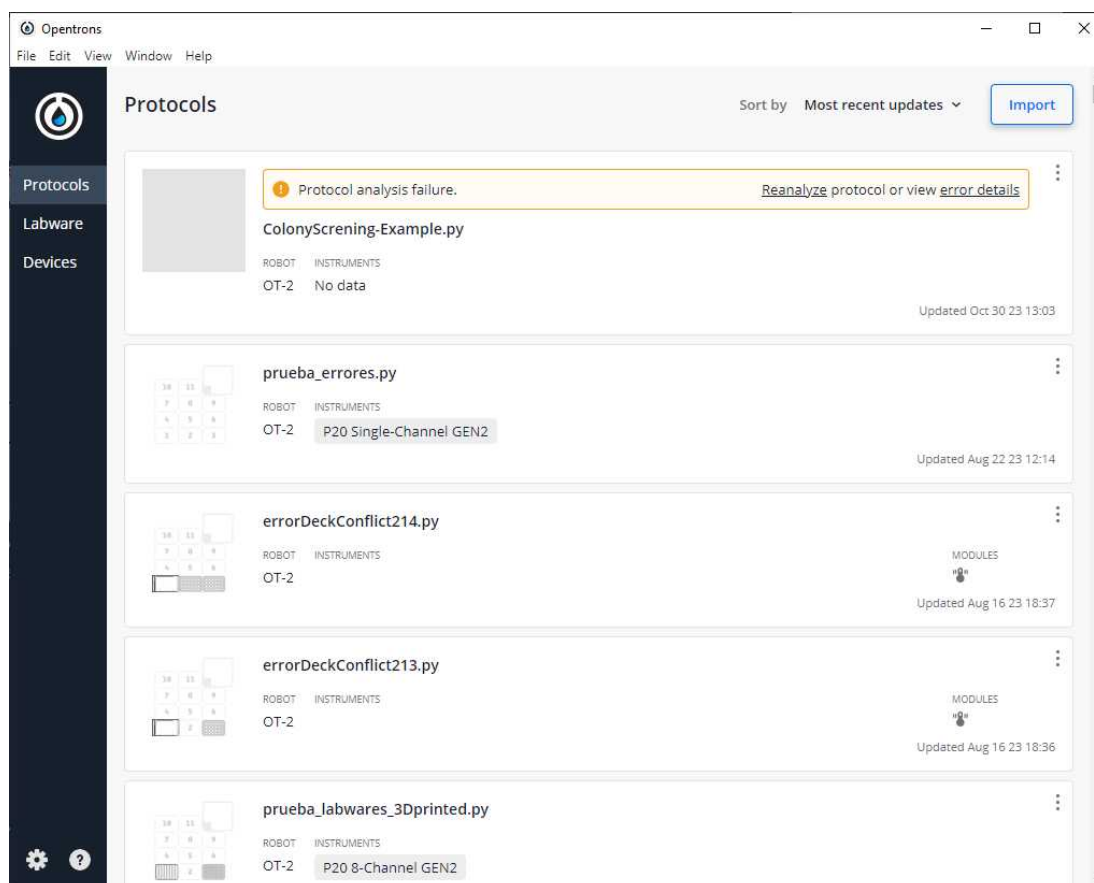
 Result of importing the Python script in the OT-App

As we can see, we have an error, but that is programmed because the script is meant to work in the robot but not in your computer

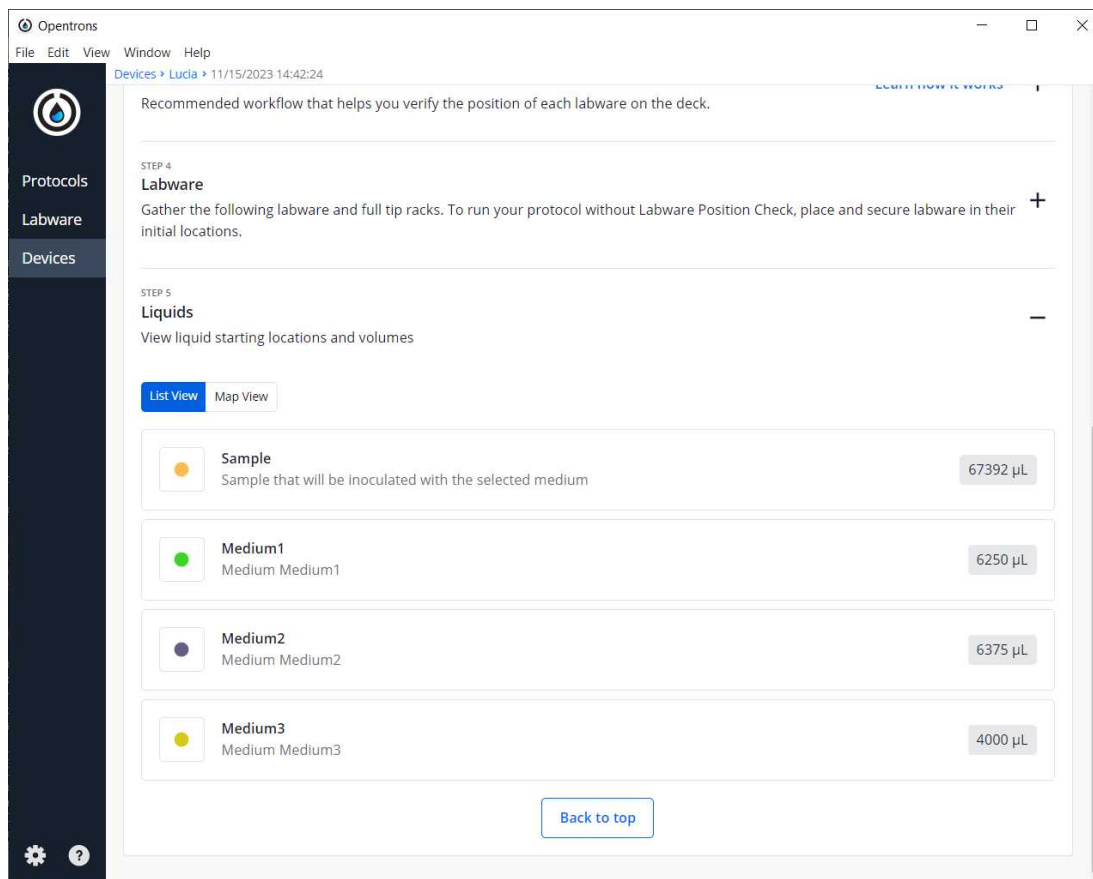**9.6** Run the protocol in the robot that we have transferred the Excel file `1m`

ColonyScreening-Example.py -> **Start setup** -> **Select robot** in which we are going to run the protocol

If we do not have any errors, the output should look similar to the following pictures
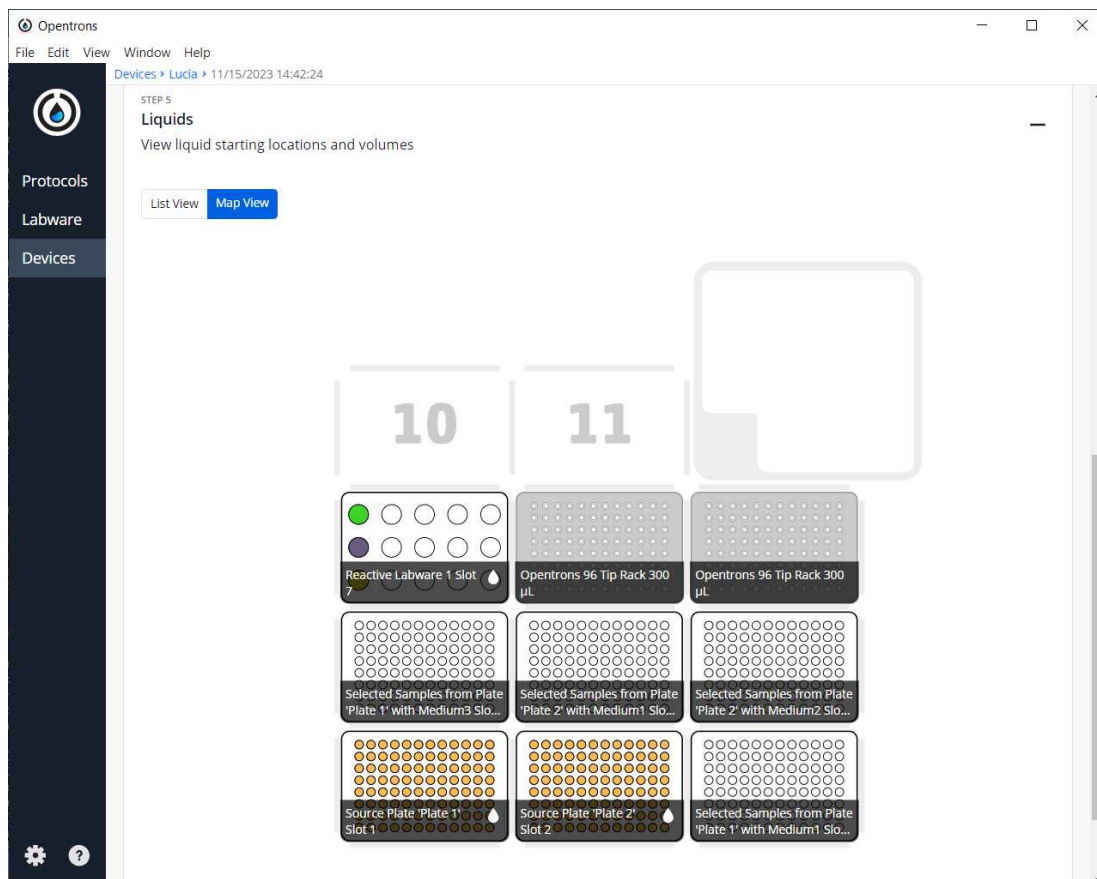
Labware and liquid set-up layout

Volumes of the needed liquids to perform the protocol

9.7 Turn the HEPA filter module   `30s`

✱

9.8 Clean platform of the robot that we are going to perform the protocol   `2m`
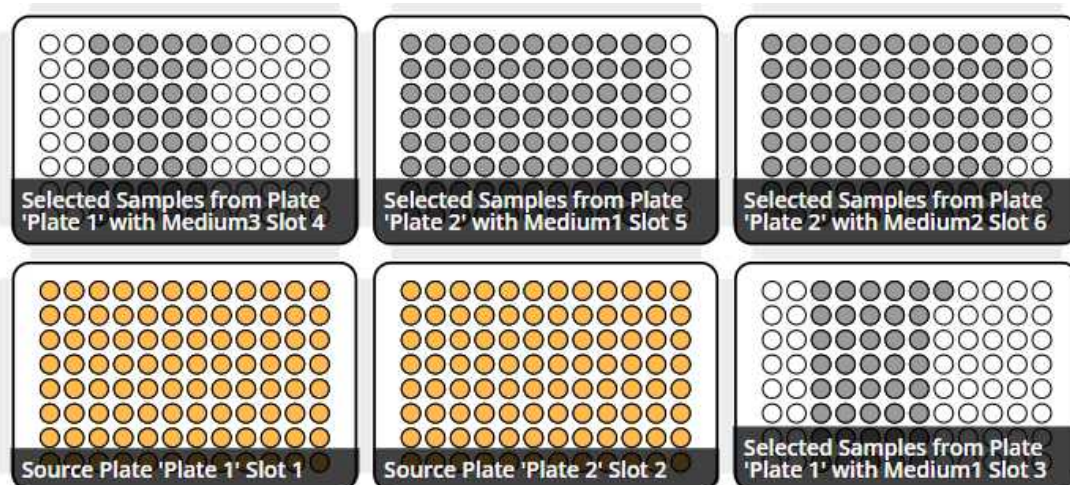
9.9 Prepare all reagents and labware in the places the App is showing and taking into account the note   `5m`
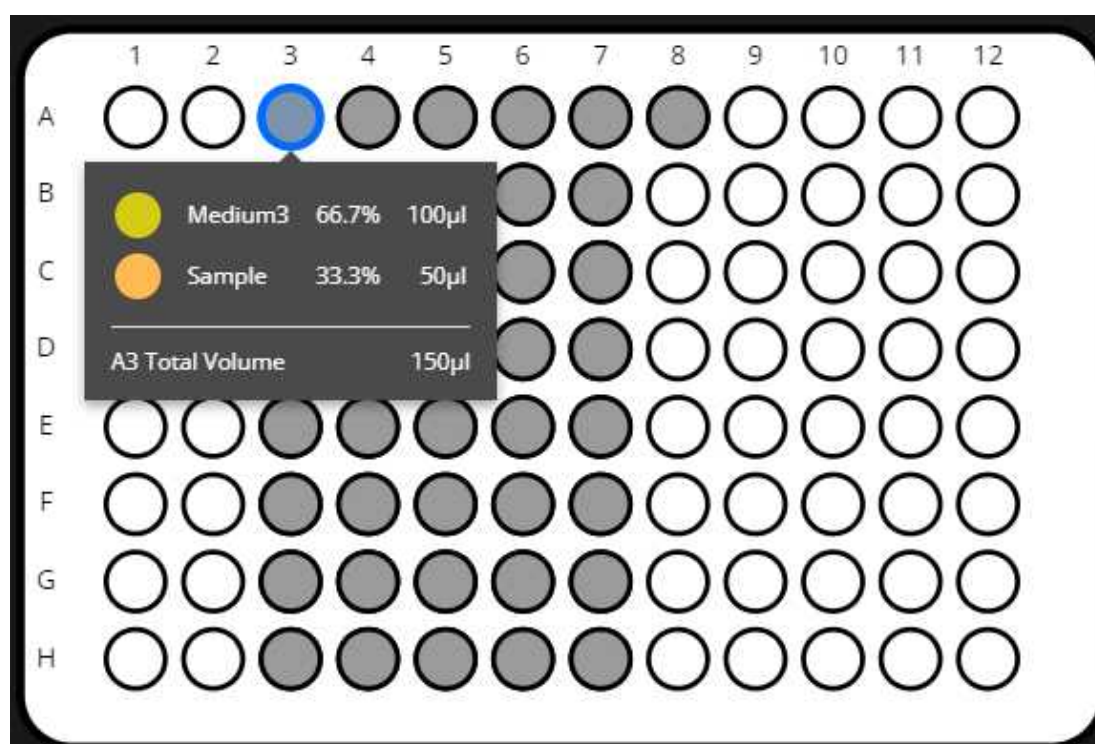in step 〉⤺ go to step #5.2 Notes

9.10 Start run   `1h 22m`

> **Expected result**

Final layout of source and final plates in the run

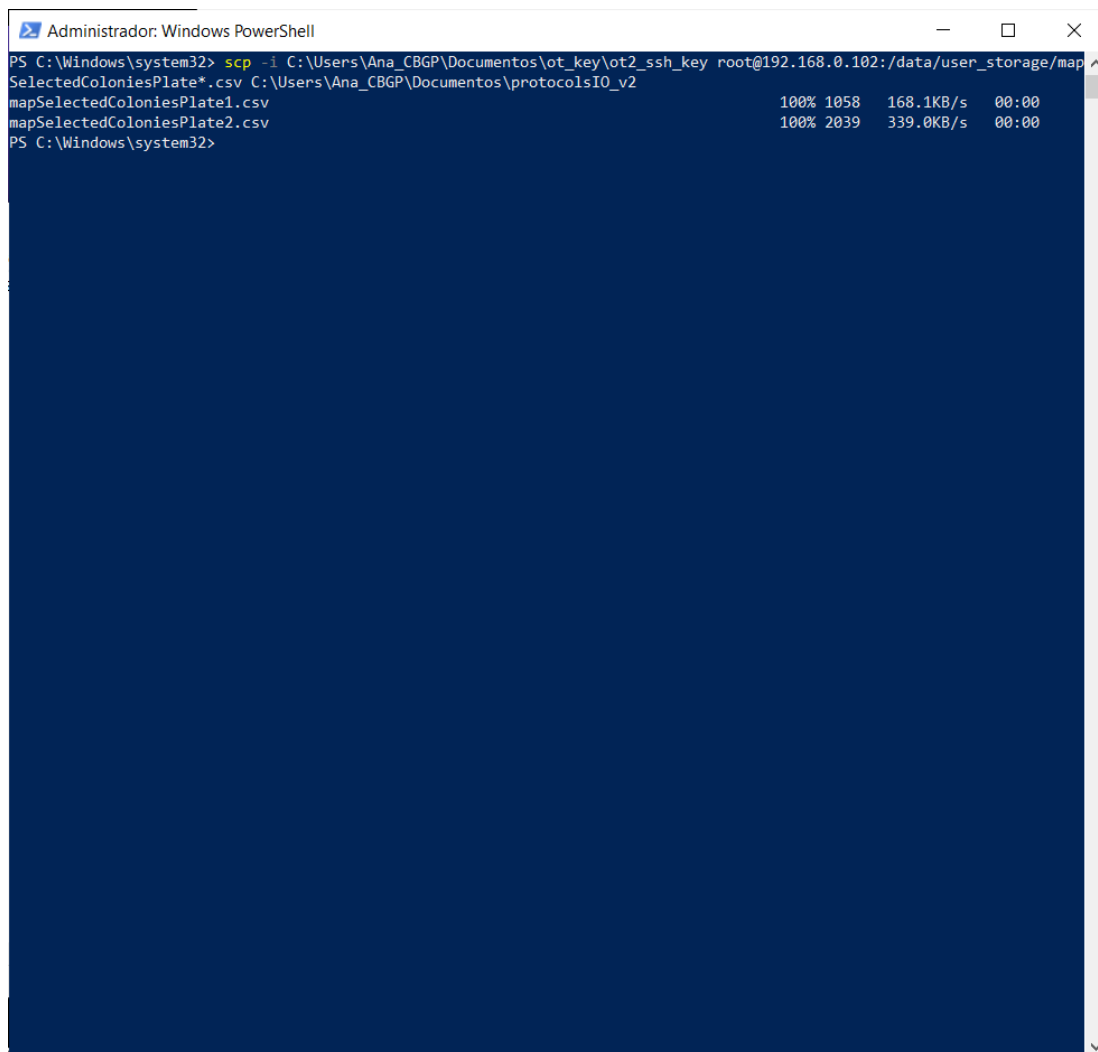Example of the content of A1 in the labware *Selected Samples from Plate 'Plate 1' with Medium3 Slot 4*

Here, we will obtain the mix between the volume of media and the samples set in the variable file in the final plates. These positions are seen in the image by the grey wells, and we can see the info of the plate and the media in the name of the labware.

9.11 Retrieve labwares from the OT `5m`

9.12 Retrieve the final maps, in this case, they will be called *mapSelectedColoniesPlate1 and mapSelectedColoniesPlate2,* of the IDs of the samples that fulfilled the requisites that have the names set in the variable file by us `2m`

command line windows with the transfer command of the samples map from the OT to our computer

| | mapSelectedColoniesPlate1.csv 1KB |
| | mapSelectedColoniesPlate2.csv 1KB |