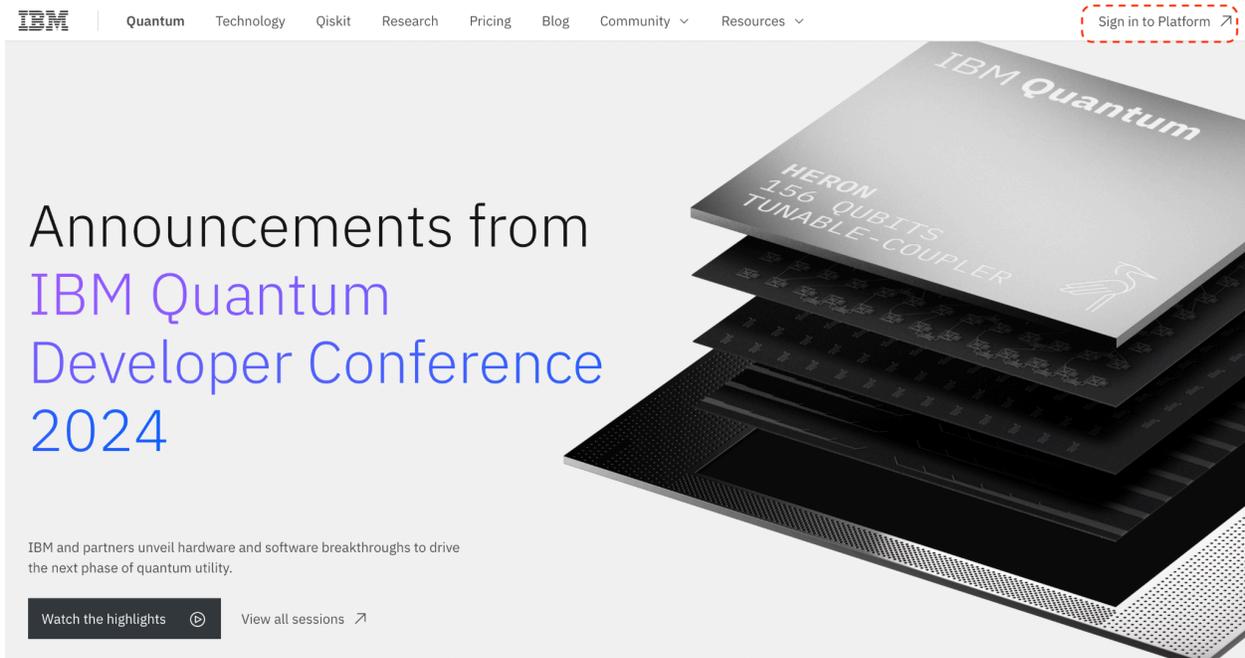
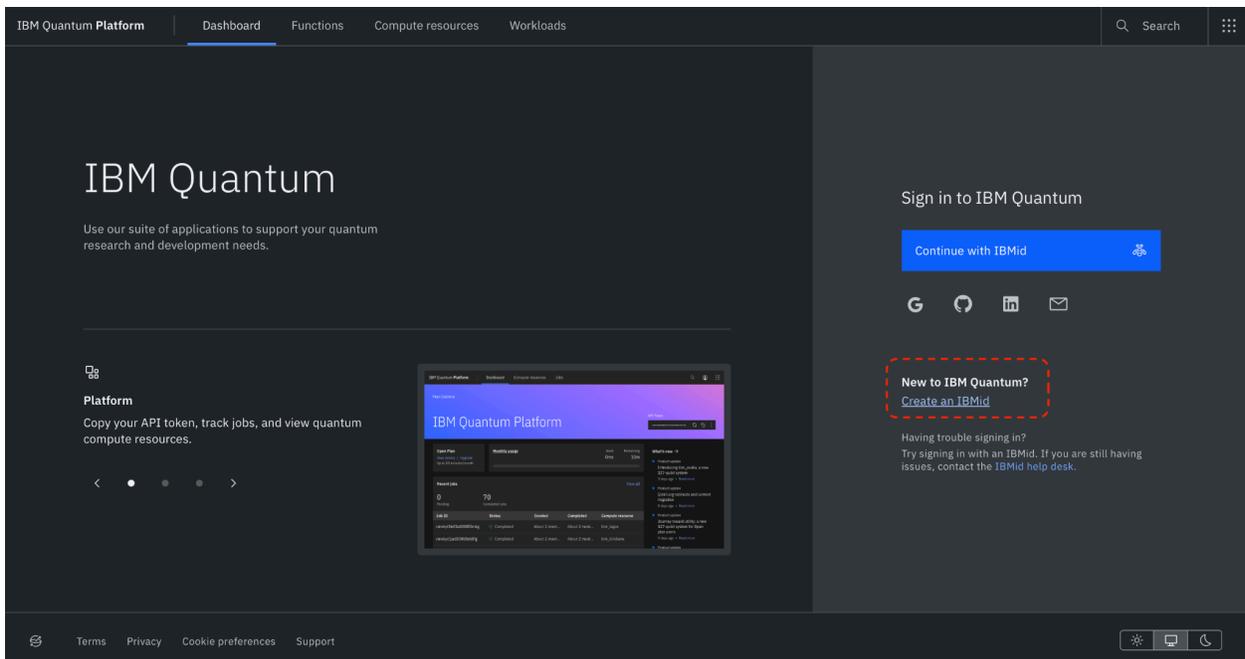


This tutorial will walk you through the sign-up process for IBM quantum. To start navigate to <https://www.ibm.com/quantum> and click on “Sign in to Platform” in the upper right corner.



It'll open a new tab so make sure to disable any pop-up blockers that might stop the link from working. Next if you don't have one click on “Create an IBMid” and follow the on screen prompts, otherwise sign in to your existing account.



After signing in your Dashboard should look like the one below.

The screenshot shows the IBM Quantum Platform dashboard for user Justin Beltran. The top navigation bar includes 'Dashboard', 'Functions', 'Compute resources', and 'Workloads'. The main header displays the user's name and the 'API Token' field, which is circled in red. Below the header, there are sections for 'Open Plan', 'Monthly usage' (0ms used, 10m remaining), and 'Recent workloads'. A table lists recent workloads with columns for ID, Status, Completed, Mode, Compute resource, and Usage. A notification pop-up is visible over the table, stating: 'The IBM Quantum Signup requirements have changed. Users must provide citizenship information.'

ID	Status	Completed	Mode	Compute resource	Usage
cx4aqyqrkac00085mh4g	Completed	28 Nov 2024	Job	ibm_sherbrooke	7s
cx4anddtpsjg008v7hkg	Completed	28 Nov 2024	Job	ibm_sherbrooke	
cx4an3cpx23g008e5pdg	Completed	28 Nov 2024	Job	ibm_sherbrooke	
cx4amsvbqkhg008xc0g0	Completed	28 Nov 2024	Job	ibm_sherbrooke	
cx4amg2pijw30008kgv5g	Completed	28 Nov 2024	Job	ibm_sherbrooke	2s

Click on copy to clipboard icon (circled in red) where the API token is located and paste into the jupyter notebook file where it says “your token here” just below the cell titled “On Quantum Computer.”

The screenshot shows a Jupyter Notebook cell titled "On a Quantum Computer". The cell contains the following Python code:

```
[ ] service = QiskitRuntimeService(channel="ibm_quantum",
                                  token="your token here",
                                  )

[ ] service.backends(min_num_qubits=6)

[ ] <IBMQBackend('ibm_brisbane')>,
    <IBMQBackend('ibm_kyiv')>,
    <IBMQBackend('ibm_sherbrooke')>]
```

The code defines a QiskitRuntimeService object with the channel "ibm\_quantum" and the token "your token here". It then lists the available backends for the service, which include 'ibm\_brisbane', 'ibm\_kyiv', and 'ibm\_sherbrooke'. The cell also includes instructions on how to use these backends, with an example using 'ibm\_sherbrooke'.

Now any cell that runs the “backend\_real” will automatically use the quantum computer backend specified in “service.backend()” but it will also debit time to run circuits against your API token and you only have 10 minutes total per month.