

14

Para saber mais: Removendo serviços

Importante

Agora que finalizamos o projeto, é importante nós removermos as máquinas e os serviços que configuramos na Amazon com o objetivo de não ultrapassar o consumo do limite gratuito. Vamos começar removendo o grupo de auto escalonamento, para isso, na aba **Services** pesquise pelo **EC2**. No menu lateral esquerdo na aba **Auto Scaling** clique na opção **Auto Scaling Groups**.

The screenshot shows the AWS Auto Scaling Groups management interface. At the top, there's a blue button for 'Create Auto Scaling group' and an 'Actions' dropdown menu. The 'Actions' menu has 'Edit' and 'Delete' options, with 'Delete' being the one currently highlighted. Below the menu, there's a search bar labeled 'Filter: Filter Auto Scaling' and some filter dropdowns for 'Name', 'Launch Configuration /', 'Instances', and 'Desired'. A table lists one item: 'Escalonando-s...' with 'imagem-customizada-c...' as its launch configuration, and both 'Instances' and 'Desired' counts set to 0.

Ao fazermos isso, as duas instâncias que tínhamos e faziam parte desse grupo de auto escalonamento deverão ter sido removidas, para confirmar, no menu lateral esquerdo, na opção **Instances** clique em **Instances** e confirme que as instâncias encontram-se no estado **Shutting down / Terminate**. Poderá levar alguns minutos para que as instâncias sejam completamente removidas da listagem:

The screenshot shows the AWS EC2 Instances management interface. The left sidebar has a 'INSTANCES' section with 'Instances' selected, indicated by a red box. Other options include 'Launch Templates', 'Spot Requests', 'Reserved Instances', 'Dedicated Hosts', and 'Scheduled Instances'. The main content area has a 'Launch Instance' button at the top. Below it, a message says 'You do not have any running instances in this region.' and 'First time using EC2? Check out the Getting Started Guide.' There's also a link to 'Click the Launch Instance button to start your own server.' and a 'Launch Instance' button at the bottom right.

Na sequência, vamos remover o平衡ador de cargas. No menu lateral esquerdo, na aba **Load Balancing**, clique em **Load Balancers** e posteriormente vá no botão **Actions** e selecione **Delete** para remover o balanceador de cargas que criamos.

The screenshot shows the AWS Load Balancers management interface. The left sidebar has a 'LOAD BALANCING' section with 'Load Balancers' selected, indicated by a red box. Other options include 'Target Groups'. The main content area has a 'Create Load Balancer' button at the top. Below it, there's a table with one item: 'balanceador-casadocodigo...' with 'casadocodigo-' as its name. An 'Actions' dropdown menu is open over this item, showing options like 'Delete', 'Edit health check', 'Edit subnets', 'Edit IP address type', 'Edit instances', 'Edit listeners', and 'Edit security groups', with 'Delete' being the highlighted option.

Na sequência, vamos remover o banco de dados, para isso, na aba **Services** procure pelo serviço do **RDS** e no menu lateral esquerdo clique na aba **Instances**

The screenshot shows the AWS RDS Dashboard. On the left sidebar, under the 'Instances' heading, there are several options: Clusters, Performance Insights (with a 'PREVIEW' badge), Reserved Instances, Snapshots, Parameter Groups, External Licenses, and Option Groups. The 'Instances' option is highlighted with a red box. On the right side, there's a promotional message about the new RDS console look, followed by an 'Amazon Aurora' section with buttons for 'Launch Aurora MySQL' and 'Launch Aurora PostgreSQL', and a link to 'Or, Restore Aurora DB Cluster from S3'.

Posteriormente clique em **Instances Actions** e clique em **Delete**

The screenshot shows the AWS RDS Dashboard. The 'Instances' section is selected. In the center, there's a table for a MySQL instance named 'banco-casadocodigo'. The 'Endpoint' is listed as 'banco-casadocodigo.ca3wyfjmzjsj.us-east-1'. Below the table, there's a section titled 'Alarms and Recent Events' with two entries. On the right, a dark gray 'Instance Actions' dropdown menu is open, showing various options like 'See Details', 'Create Read Replica', 'Promote Read Replica', etc., with 'Delete' highlighted.

Ao fazermos isso, a Amazon vai perguntar se queremos fazer uma cópia dos dados presentes no Banco, colocamos **No** e confirmamos a deleção.

The screenshot shows the 'Delete DB Instance' dialog box. It asks if you're sure you want to delete the 'banco-casadocodigo' DB Instance. There are two dropdowns for 'Create final Snapshot': 'No' and 'Yes'. A checkbox below says 'I acknowledge that upon instance deletion, automated backups, including system snapshots and point-in-time recovery, will no longer be available.' A warning message at the bottom states: 'We strongly recommend taking a final snapshot before instance deletion since after your instance is deleted, automated backups will no longer be available.' At the bottom right are 'Cancel' and 'Delete' buttons.

Para finalizar, vamos remover o Redis que configuramos, para isso, vá até a aba **Services** e procure pelo serviço do **Elasticache** e no menu lateral esquerdo clique em **Redis**

The screenshot shows the AWS ElastiCache Dashboard. On the left sidebar, under the 'Redis' section, the 'Redis' option is highlighted with a red box. The main content area displays a list of resources: Memcached (0), Redis (1), Reserved Nodes (0), Backups (0), Manual (0), Automated (0), and Recent Events (3). A button labeled 'Request cache node limit increase' is visible at the top right.

Posteriormente, clique no botão **Delete** para removermos o Redis:

The screenshot shows the AWS ElastiCache Dashboard with the Redis section selected. The 'Delete' button in the top navigation bar is highlighted with a red box. Below, a table lists a single Redis cluster named 'redis-casadocodigo' with Mode set to 'Redis', 0 shards, and 1 node.

Cluster Name	Mode	Shards	Nodes
redis-casadocodigo	Redis	0	1