

API Microsoft Azure

- BLOBs
 - Criando um blob (*arquivo*), API HTTP:

HTTP método PUT em

<https://myaccount.blob.core.windows.net/mycontainer/myblob>

O conteúdo enviado pelo método PUT será o arquivo.

A chamada HTTP é autenticada usando credenciais do administrador da conta Azure.

API Microsoft Azure

- BLOBs
 - Criando um blob usando API Python (que usa a API HTTP):

```
from azure.storage import BlobService

service = BlobService(account_name='the_account_name',
                      account_key='senha_de_autenticacao')

# Criar um BLOB chamado arq.zip em um container fazendo upload do arquivo
# em /tmp
service.put_block_blob_from_path("container_name", "arq.zip", "/tmp/arq.zip")
```

API Microsoft Azure

- Queues
 - Criando uma fila (SDK Python)

```
from azure.storage import QueueService
```

```
# Criar nova fila no serviço de armazenamento
```

```
queue_service = QueueService(account_name='my_login', account_key='my_key')
```

```
queue_service.create_queue('new_queue_name')
```

API Microsoft Azure

- Queues
 - Acessando uma fila (SDK Python)

```
from azure.storage import QueueService
```

```
# Gravar uma mensagem em uma fila
```

```
queue_service = QueueService(account_name='my_login', account_key='my_key')
```

```
queue_service.put_message('new queue name', 'message payload')
```

API Microsoft Azure

- Queues
 - Acessando uma fila (SDK Python)

```
from azure.storage import QueueService
```

```
# Recuperar as primeiras mensagens de uma fila, removendo-as da fila.
```

```
queue_service = QueueService(account_name='my_login', account_key='my_key')
```

```
messages = queue_service.get_messages('new queue name')
```

```
for m in messages:
```

```
    print(m.message_text)
```

```
    queue_service.delete_message('new queue name', m.message_id, m.pop_receipt)
```

API Microsoft Azure

- Máquinas Virtuais
 - Criando uma máquina virtual (Python)

```
from azure import *
from azure.servicemanagement import *

# A API REST de gestão de máquinas virtuais requer autenticação com chave privada SSL
sms = ServiceManagementService(subscription_id, certificate_path)

name = 'vm_name'
location = 'West US'

# Criar o serviço
sms.create_hosted_service(service_name=name, label=name, location=location)

# Imagem de disco a ser usada para criar a máquina virtual
image_name = 'OpenLogic__OpenLogic-CentOS-62-20120531-en-us-30GB.vhd'

# Conta de armazenamento onde o HD será guardado
media_link = 'url_to_target_storage_blob_for_vm_hd'
(...)
```

API Microsoft Azure

- Máquinas Virtuais
 - Criando uma máquina virtual (Python)

```
# Facilitador para configuração de máquina Linux
linux_config = LinuxConfigurationSet('myhostname', 'myuser', 'mypassword', True)

os_hd = OSVirtualHardDisk(image_name, media_link)

# Executa a criação da VM propriamente dita
sms.create_virtual_machine_deployment(service_name=name,
    deployment_name=name,
    deployment_slot='production',
    label=name,
    role_name=name,
    system_config=linux_config,
    os_virtual_hard_disk=os_hd,
    role_size='Small')
```