

# Controle das Tarefas do Shell

---

*Comandos `bg`, `fg` e `jobs`*

# Controle de Tarefas

.....

Habilidade do shell de interromper e retomar a execução de comandos, além de poder colocá-los no background para serem executados em segundo plano.

Já o programa em foreground é aquele que está vinculado ao terminal.

Quando um processo é colocado em background, criamos uma tarefa. Cada tarefa possui um número de identificação, contando a partir de 1 (não confundir com PID de processos).

Colocamos um processo em background para liberar o terminal de onde ele foi iniciado.

Programas não-interativos são colocados em background adicionando-se um `&` logo após o comando.

# Controle de Tarefas

.....

Exemplo:

`gimp &`

Para colocá-lo em background, digite Ctrl+Z (sinal TSTP, 20), e em seguida digite o comando **bg** para reiniciar a tarefa em background.

Para colocar a tarefa novamente em foreground, digite o comando **fg**.

Exemplo:

`vi /home/fabio/arq1`

`Ctrl+Z`

`bg`

#Agora a tarefa está em background. Visualize com ps, e use o terminal para #digitar outros comandos. Para voltar a tarefa ao foreground então:

`fg`

# Comando jobs

.....

Lista as tarefas ativas.

jobs [opções]

-l Lista também os PIDs.

# bg

.....

Coloca a tarefa especificada em background (equivale a iniciar o programa com &). Se a tarefa não for especificada, será usada a tarefa atual do shell. A tarefa deve estar pausada antes (Ctrl+Z).

Sintaxe:

`bg [tarefa]`

# fg



Coloca a tarefa especificada em foreground, transformando-a na tarefa atual. Se a tarefa não for especificada, será usada a tarefa atual do shell.

Sintaxe

fg [tarefa]

# Exemplo completo

.....

#Abrir vi e colocá-lo em background:

vi teste

Ctrl+Z|

bg

# Verificar jobs

jobs

#Abrir gimp em background:

gimp &

#Verificar jobs:

jobs

#Pausar gimp com sinal TSTP:

ps

kill -20 [PID do gimp]

#verificar jobs novamente

jobs

#Trazer processos ao foreground e finalizá-los (fechar normalmente):

fg [tarefa]