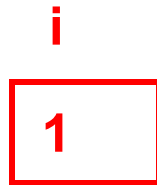


`int i = 1;` ← **Vater** erzeugt und initialisiert Variable *i*

```
if (fork()==0) {  
    printf("i im Sohn: %d\n",i);  
    i = 2;  
    printf("i im Sohn: %d\n",i);  
    exit(0);  
}  
  
sleep(2);  
printf("i im Vater: %d\n",i);
```



```
int i = 1;
```

```
if (fork()==0) {  fork(): Sohn entsteht
```

```
    printf("i im Sohn: %d\n",i);
```

```
    i = 2;
```

```
    printf("i im Sohn: %d\n",i);
```

```
    exit(0);
```

```
}
```

```
sleep(2);
```

```
printf("i im Vater: %d\n",i);
```

i (Vater)



i (Sohn)



*fork() kopiert den **Vater**:
Sohn erhält eigene Variable *i*
Anfangswert wird aus der
Variablen des Vaters übernommen*

```
int i = 1;
if (fork()==0) {
    printf("i im Sohn: %d\n", i);
    i = 2;
    printf("i im Sohn: %d\n", i);
    exit(0);
}

sleep(2);
printf("i im Vater: %d\n", i);
```

i (Vater)

1

i (Sohn)

1

```
int i = 1;
```

```
if (fork()==0) {
```



```
    printf("i im Sohn: %d\n", i);
```

```
    i = 2;
```

```
    printf("i im Sohn: %d\n", i);
```

```
    exit(0);
```

```
}
```

```
sleep(2);
```



Vater schläft zunächst

```
printf("i im Vater: %d\n", i);
```

i (Vater)

1

i (Sohn)

1

```
int i = 1;
```

```
if (fork()==0) {
```

```
    printf("i im Sohn: %d\n", i);
```

```
    i = 2;
```

```
    printf("i im Sohn: %d\n", i);
```

```
    exit(0);
```

```
}
```

```
sleep(2);
```

```
printf("i im Vater: %d\n", i);
```

← **Sohn-Ausgabe: 1**

i (Vater)

1

i (Sohn)

4 2

```
int i = 1;
```

```
if (fork()==0) {
```

```
    printf("i im Sohn: %d\n", i);
```

```
    i = 2; ← Sohn ändert Inhalt seiner Variablen
```

```
    printf("i im Sohn: %d\n", i);
```

```
    exit(0);
```

```
}
```

```
sleep(2); ←
```

```
printf("i im Vater: %d\n", i);
```

i (Vater)

1

i (Sohn)

2

```
int i = 1;
```

```
if (fork()==0) {
```

```
    printf("i im Sohn: %d\n", i);
```

```
    i = 2;
```

```
    printf("i im Sohn: %d\n", i);
```

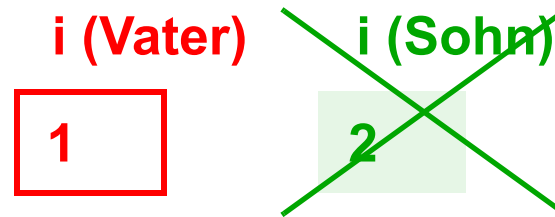
```
    exit(0);
```

```
}
```

```
sleep(2);
```

```
printf("i im Vater: %d\n", i);
```

← **Sohn-Ausgabe: 2**



```
int i = 1;
if (fork()==0) {
    printf("i im Sohn: %d\n",i);
    i = 2;
    printf("i im Sohn: %d\n",i);
    exit(0); ← Sohn beendet sich, Variable wird gelöscht
}

sleep(2); ←
printf("i im Vater: %d\n",i);
```


i (Vater)

1

```
int i = 1;
if (fork()==0) {
    printf("i im Sohn: %d\n",i);
    i = 2;
    printf("i im Sohn: %d\n",i);
    exit(0);
}
sleep(2);
printf("i im Vater: %d\n",i);
```

Vater-Ausgabe seiner Variablen: 1