

Faça o que eu fiz na aula

Vamos fazer um método `factory` juntos! Para criar um padrão `factory`, precisamos entender com o que estamos trabalhando e como vamos saber qual a hora certa de chamar uma classe ou outra. Vamos utilizar o mesmo exemplo da aula e trabalhar com documentos.

1. Vamos decidir como vamos selecionar uma classe ou outra:

A classe CPF será chamada quando o documento conter **11** dígitos, e a classe CNPJ quando possuir **14**.



1. Criamos a classe e definimos um método estático que decidirá qual classe instanciar:

```
class Documento:
    @staticmethod
    def criar_novo(documento):
```

1. Agora, criamos os condicionais:

```
class Documento:
    @staticmethod
    def criar_novo(documento):
        doc_str = str(documento)
        if len(doc_str) == 11:
            return DpcCpf(doc_str)
        if len(doc_str) == 14:
            return DpcCnpj(doc_str)
```

1. Depois disso, fica fácil incluir uma opção nova de classe:

```
class Documento:
    @staticmethod
    def criar_novo(documento):
        doc_str = str(documento)
        if len(doc_str) == 11:
            return DpcCpf(doc_str)
        if len(doc_str) == 14:
            return DpcCnpj(doc_str)
        if len(doc_str) == 20:
            return DocQualquer(doc_str)
```

1. Por último, vamos configurar uma exceção para quando o documento não for de nenhum tipo:

```
class Documento:
    @staticmethod
    def criar_novo(documento):
        doc_str = str(documento)
```

```
if len(doc_str) == 11:
    return DpcCpf(doc_str)
if len(doc_str) == 14:
    return DpcCnpj(doc_str)
if len(doc_str) == 20:
    return DocQualquer(doc_str)
Raise ValueError("Documento incorreto!")
```