5 Tipos de Serviços

**Provider**

Serviço que vai criar um serviço, “Armazene configuração que fornece para nosso serviço na nossa aplicação”

<https://docs.angularjs.org/guide/providers>

**Value Recipe**

(Armazenar informações pura e podemos modificar o valor no meio da execução)

Digamos que nós queremos ter um serviço muito simples chamado "clientId" que fornece uma cadeia que representa um ID de autenticação usado por algum API remoto. Você poderia defini-lo como este:

**Constant Recipe**

(Armazenar as informações e ela não muda durante o tempo de execução).

**Service**

**Desenvolvedores de JavaScript costumam usar tipos personalizados para escrever código orientado a objeto. Vamos explorar como nós poderíamos lançar um unicórnio para o espaço através do nosso unicornLauncherserviço que é uma instância de um tipo customizado:**

**Factories**

Receita fábrica

A receita Valor é muito simples de escrever, mas carece de algumas características importantes que muitas vezes precisa ao criar serviços. Vamos agora olhar para o irmão mais poderoso da receita Valor, o Factory. A receita de fábrica adiciona as seguintes habilidades:

capacidade de utilizar outros serviços (têm dependências)

inicialização do serviço

atrasou a inicialização / preguiçoso

A receita da fábrica constrói um novo serviço usando uma função com zero ou mais argumentos (estes são dependências de outros serviços). O valor de retorno desta função é a instância do serviço criado por esta receita.

Nota: Todos os serviços em angular são singletons. Isso significa que o injector utiliza cada receita no máximo uma vez para criar o objecto. O injector, em seguida, armazena em cache a referência para todas as necessidades futuras.

Uma vez que uma fábrica é uma versão mais potente do Valor receita, o mesmo serviço pode ser const