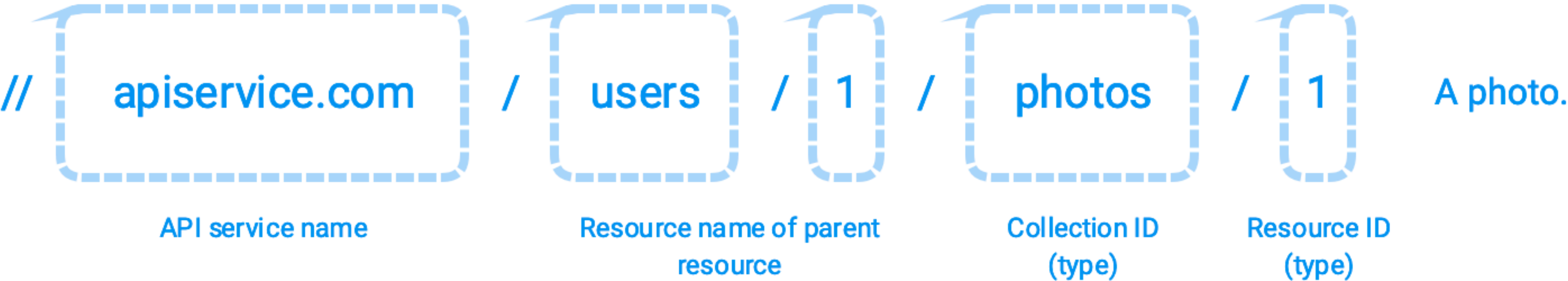
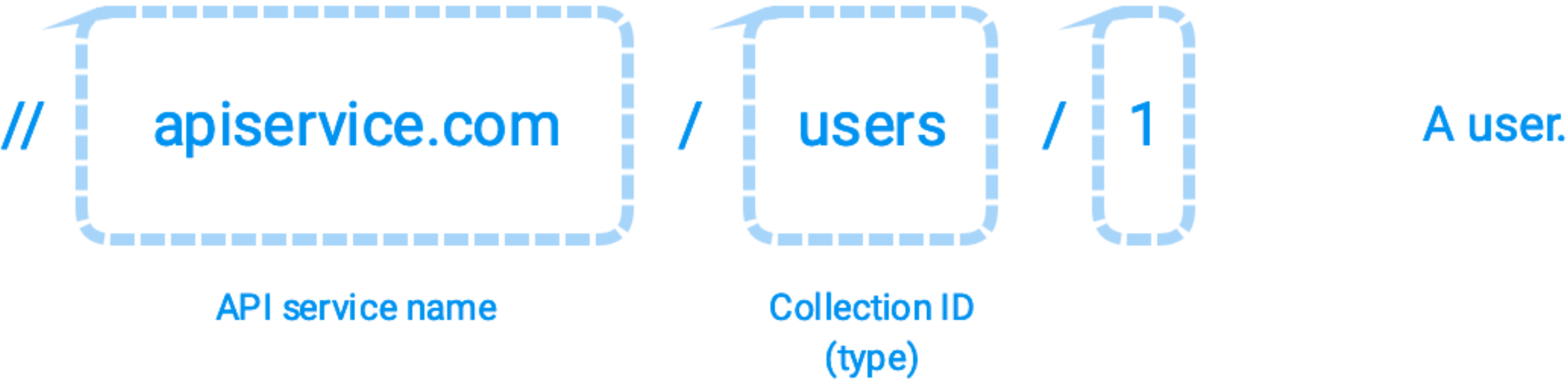


DRF ViewSets & CRUD

Django Development II

Azimjon Pulatov | PDP Online

Rest API Design Qilish



Method	HTTP Request Method (Verb)
LIST	GET
GET	GET
CREATE	POST
UPDATE	PATCH
DELETE	DELETE

Todo App API

Task	Method	Path
Create a new task	POST	/tasks
Delete an existing task	DELETE	/tasks/{id}
Get a specific task	GET	/tasks/{id}
Search for tasks	GET	/tasks
Update an existing task	PUT	/tasks/{id}

ModelViewSet = Set of Views

- Create View
- Retrieve View
- Update View
- List View
- Destroy View
- ...

DRF ViewSet Mixins

```
class ModelViewSet(mixins.CreateModelMixin,
                   mixins.RetrieveModelMixin,
                   mixins.UpdateModelMixin,
                   mixins.DestroyModelMixin,
                   mixins.ListModelMixin,
                   GenericViewSet):
    """
    A viewset that provides default `create()`,
    `retrieve()`, `update()`, `partial_update()`,
    `destroy()` and `list()` actions.
    """
    pass
```


DRF ViewSets

```
# views.py
```

```
from rest_framework.viewsets import ModelViewSet
```

```
from .models import Book
```

```
from .serializers import BookSerializer
```

```
class BookViewSet(ModelViewSet):
```

```
    queryset = Book.objects.all()
```

```
    serializer_class = BookSerializer
```

```
GET /books/
```

```
GET /books/{id}/
```

```
POST /books/
```

```
PUT /books/{id}/
```

```
PATCH /books/{id}/
```

```
DELETE /books/{id}/
```


DRF ViewSet Actions

url = **GET** /books/featured/

```
from rest_framework.decorators import action

class BookViewSet(ModelViewSet):
    queryset = Book.objects.all()
    serializer_class = BookDetailSerializer

    @action(detail=False, methods=["get"])
    def featured(self, request):
        books = self.get_queryset().filter(featured=True)
        serializer = self.get_serializer(books, many=True)
        return Response(serializer.data, status=status.HTTP_200_OK)
```

DRF Routers

```
from django.urls import path, include
from rest_framework.routers import DefaultRouter
from snippets import views

# Create a router and register our viewsets with it.
router = DefaultRouter()
router.register(r'books', views.BookViewSet)

# The API URLs are now determined automatically by the router.
urlpatterns = [
    path('', include(router.urls)),
]
```


Let's Code

