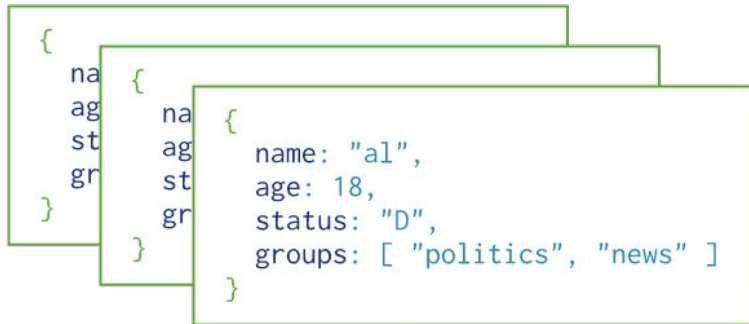


# Введение в



# MongoDB

Document Oriented!



The image shows three overlapping rectangular boxes, each containing a snippet of a JSON document. The boxes are arranged from top-left to bottom-right, with each subsequent box being slightly offset to the right and bottom. The top box shows the opening curly brace and the first few fields: 'na', 'ag', 'st', and 'gr'. The middle box shows the opening curly brace and the first few fields: 'na', 'ag', 'st', and 'gr'. The bottom box shows the full document structure with fields: 'name: "a1"', 'age: 18', 'status: "D"', and 'groups: [ "politics", "news" ]'.

```
{  
  na  
  ag  
  st  
  gr  
}
```

```
{  
  na  
  ag  
  st  
  gr  
}
```

```
{  
  name: "a1",  
  age: 18,  
  status: "D",  
  groups: [ "politics", "news" ]  
}
```

# MongoDB

Document Oriented!

```
{
  name: "al",
  age: 18,
  status: "D",
  groups: [ "politics", "news" ]
}
```

**V**  
**S**

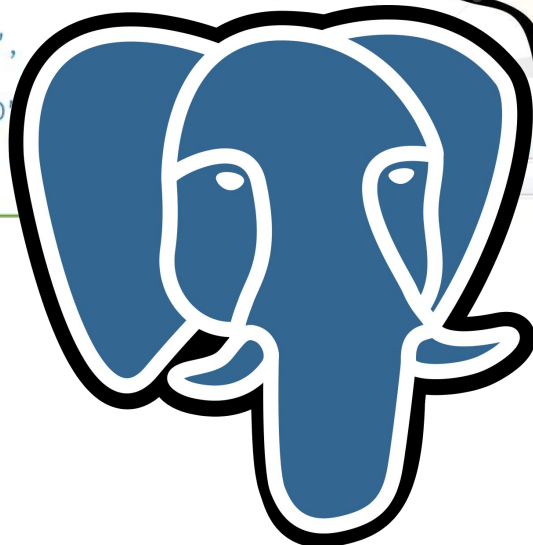
	id	emp_name	emp_age	emp_salary	join_date	phone
1	1	Mike	35	5000.50	2016-01-01 00:00:00.000	042-4521-458452
2	2	Michale	30	4500.00	2016-05-01 00:00:00.000	021-1234-574845
3	3	Jimmy	27	3000.75	2016-05-03 00:00:00.000	001-4521-458452
4	4	Shaun	30	3500.30	2018-01-20 00:00:00.000	042-2544-458452
5	5	Ben	45	5000.00	2015-03-15 00:00:00.000	001-3355-457484
6	6	John	28	2600.00	2018-04-12 19:35:27.067	092-1414-458452
7	7	NULL	40	5500.00	2018-03-20 00:00:00.000	042-3333-458452
8	8	Jay	36	5500.00	2018-03-25 00:00:00.000	021-4242-145854
9	9	Haynes	25	3300.00	2018-03-20 00:00:00.000	033-1133-457457

Relational

# MongoDB

Document Oriented!

```
{
  name: "al",
  age: 18,
  status: "D",
  groups: [
  ]
}
```



		phone				
	00:00:00.000					042-4521-458452
	00:00:00.000					021-1234-574845
	00:00:00.000					001-4521-458452
	00:00:00.000					042-2544-458452
	00:00:00.000					001-3355-457484
	19:35:27.067					092-1414-458452
/	NULL	40	5500.00	2018-03-20 00:00:00.000		042-3333-458452
8	Jay	36	5500.00	2018-03-25 00:00:00.000		021-4242-145854
9	Haynes	25	3300.00	2018-03-20 00:00:00.000		033-1133-457457

Not Relational!

# MongoDB

## Document Oriented!

```
db.user.aggregate([
  { $lookup: {
    from: "user2permission",
    localField: "_id",
    foreignField: "_id.user_id",
    as: "permission_link"
  }},
  { $unwind: "permission_link" },
  { $lookup: {
    from: "permission",
    localField:
      "permission_link.id.permission_id",
    foreignField: "_id",
    as: "permission"
  }},
  { $unwind: "permission" }
])
```

VS

```
SELECT *
FROM user
JOIN user2permission USING (user_id)
JOIN permission USING (permission_id)
```

# MongoDB

Schema free!

The screenshot shows the MongoDB Compass interface. The top bar indicates the current database is 'urlShortener' and the collection is 'foo'. The command bar shows the query: `db.getCollection('foo').find({})`. The results pane shows two documents. The first document has fields `_id`, `foo`, and `pupa`. The second document has fields `_id`, `foo`, `pupa`, `lupa`, and `biba`. The 'Type' column shows the data types for each field. Red and blue circles highlight specific fields and their types.

Key	Value	Type
(1) ObjectId("61427b4421a76d9fc59d7292")	{ 3 fields }	Object
_id	ObjectId("61427b4421a76d9fc59d7292")	ObjectId
foo	bar	String
pupa	lupa	String
(2) ObjectId("61427b6021a76d9fc59d7293")	{ 3 fields }	Object
_id	ObjectId("61427b6021a76d9fc59d7293")	ObjectId
foo	25.0	Double
pupa	{ 1 field }	Object
lupa	{ 1 field }	Object
biba	kuka	String

# MongoDB

Schema free!

- no need in DDL scripts
- no migration tools

# MongoDB. Storage

- BSON format — binary JSON + extra types:
  - native timestamps
  - ObjectId
  - etc...
- Indices — B-Tree
  - ordered multi-key indexing
  - fast lookup by key
  - index-based ordered iteration

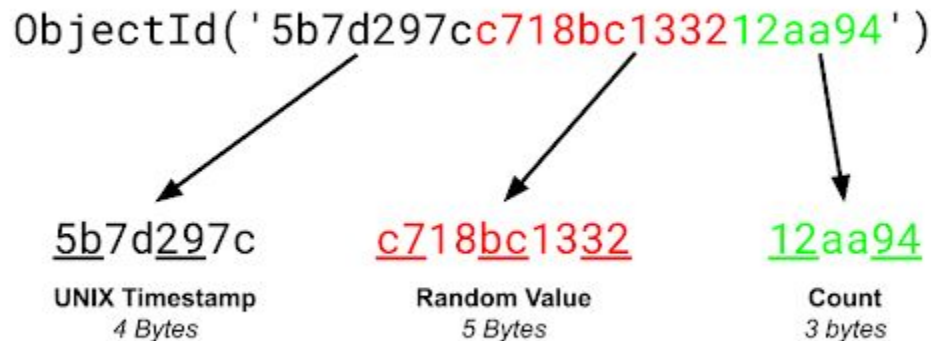


# MongoDB. Storage

- `_id` — mandatory primary key of arbitrary key

# MongoDB. Storage

- `_id` — mandatory primary key of arbitrary key
- `ObjectId` — default id type



Let's Check it Out!

