

Шпаргалка по функциям Firebase Firestore:

◆ 1. `collection(db, "students")`

Создаёт ссылку на коллекцию внутри базы Firestore.

```
const studentsCol = collection(db, "students");
```

- `db` — объект базы (`getFirestore(app)`).
- `"students"` — название коллекции.

👉 Это как путь `"db/students"`, чтобы дальше добавлять/читать документы.

◆ 2. `addDoc(collectionRef, data)`

Добавляет **новый документ** в коллекцию.

```
await addDoc(studentsCol, { name, grades: [] });
```

- `studentsCol` — ссылка на коллекцию.
 - `{ name, grades: [] }` — объект с данными.
 - Firebase сам создаст уникальный `id`.
-

◆ 3. `getDocs(collectionRef)`

Получает **все документы** из коллекции (разовый запрос, без автообновления).

Пример (в твоём коде пока не используется напрямую, только `onSnapshot`):

```
const snapshot = await getDocs(studentsCol);
```

```
snapshot.forEach(doc => console.log(doc.id, doc.data()));
```

◆ 4. `doc(db, "students", id)`

Создаёт ссылку на **конкретный документ** в коллекции.

```
const studentRef = doc(db, "students", studentId);
```

- `db` — база данных.
 - `"students"` — коллекция.
 - `studentId` — уникальный ID документа.
-

◆ 5. updateDoc(docRef, data)

Обновляет данные в существующем документе.

```
await updateDoc(doc(db, "students", this.#id), this.toJSON());
```

- doc(db, "students", this.#id) — ссылка на документ.
 - this.toJSON() — объект с новыми данными.
 - ⚠ Если поле не указано → оно остаётся без изменений.
-

◆ 6. deleteDoc(docRef)

Удаляет документ из базы.

```
await deleteDoc(doc(db, "students", this.#id));
```

👉 Весь документ с этим id будет полностью удалён.

◆ 7. onSnapshot(collectionRef, callback)

"Подписка" на изменения.

```
onSnapshot(studentsCol, (snapshot) => {
```

```
...
```

```
});
```

- Каждый раз, когда в students что-то меняется (новый документ, изменение, удаление) → автоматически вызывается callback.
 - В snapshot приходят все документы коллекции в актуальном виде.
-

✦ Итого:

- **Создание** → addDoc
- **Чтение (разовое)** → getDocs
- **Чтение (реактивное, автообновление)** → onSnapshot
- **Обновление** → updateDoc
- **Удаление** → deleteDoc