**ДЗ: API (на JS)**

|  |  |  |
| --- | --- | --- |
| **Действие** | **Код в VS** | **Запрос в Postman** |
| Получение всех жанров | **Queries.JS**  const getGenre = (request, response) => {      pool.query('SELECT \* FROM genre ORDER BY id ASC', (error, results) => {          if (error) {          throw error          }          response.status(200).json(results.rows)      });  };  module.exports = {      getBook,  **getGenre,**      getBookById,      createBook,      updateBook,      deleteBook,      deleteGenre  };  **Index.js**  app.get('/genre', db.getGenre); | GET localhost:3000/genre |
| Добавление нового жанра | **Queries.JS**  const createGenre = (request, response) => {      const { genre\_id, name\_genre, description } = request.body;      pool.query('INSERT INTO genre (genre\_id, name\_genre, description) VALUES ($1, $2, $3) RETURNING \*', [genre\_id, name\_genre, description], (error, results) => {          if (error) {              throw error          }          response.status(201).send(`Genre added with ID: ${results.rows[0].id}`)      });  };  module.exports = {      getBook,      getGenre,      getBookById,      createBook,  **createGenre**,      updateBook,      deleteBook,      deleteGenre  };  **Index.js**  app.post('/genre', db.createGenre); | POST localhost:3000/genre  {  genre\_id:5,  "name\_genre":"Роман",  "description":"Классическая отечественная проза",  }; |
| Удаление жанра | **Queries.JS**  const deleteGenre  = (request, response) => {      const id = parseInt(request.params.id);             throw error        pool.query('DELETE FROM genre WHERE id = $1', [id], (error, results) => {          if (error) {         }          response.status(200).send(`Genre deleted with ID: ${id}`)      });  };  module.exports = {      getBook,      getBookById,      createBook,      updateBook,      deleteBook,  **deleteGenre**  };  **Index.js**  app.delete('/genre/:id',db.deleteGenre); | DELETE localhost:3000/genre?id |
| Просмотр всех книг | **Queries.JS**  const getUsers = (request, response) => {  pool.query('SELECT \* FROM users ORDER BY id ASC', (error, results) => {  if (error) {  throw error  }  response.status(200).json(results.rows)  });  };  module.exports = {  **getBook,**  getBookById,  createBook,  updateBook,  deleteBook,  };  **Index.js**  app.get('/book', db.getBook) | GET localhost:3000/book |
| Добавление новой книги | **Queries.JS**  const createBook = (request, response) => {      const { book\_id, name, author, pgenre\_id } = request.body;      pool.query('INSERT INTO book (book\_id, name, author, pgenre\_id) VALUES ($1, $2, $3, $4) RETURNING \*', [book\_id, name, author, pgenre\_id], (error, results) => {          if (error) {              throw error          }          response.status(201).send(`Book added with ID: ${results.rows[0].id}`)      });  };  module.exports = {  getBook,  getBookById,  **createBook,**  updateBook,  deleteBook,  };  **Index.js**  app.post('/book', db.createBook); | POST localhost:3000/book  {  book\_id:6,  "name":"Русалочка",  " author":" Ганс Христиан Андерсен ",  pgenre\_id:1  }; |
| Удаление книги | **Queries.JS**  const deleteBook = (request, response) => {      const id = parseInt(request.params.id);      pool.query('DELETE FROM book WHERE id = $1', [id], (error, results) => {          if (error) {              throw error          }          response.status(200).send(`Book deleted with ID: ${id}`)      });  };  module.exports = {  getBook,  getBookById,  createBook,  updateBook,  **deleteBook,**  };  **Index.js**  app.delete('/book/:id',db.deleteBook); | DELETE localhost:3000/book?id |
| Обновление данных о книге | **Queries.JS**  const updateBook = (request, response) => {      const id = parseInt(request.params.id);      const { book\_id, name, author, pgenre\_id } = request.body;      pool.query(          'UPDATE book SET book\_id = $1, name = $2, author = $3, pgenre\_id = $4 WHERE id = $5',[book\_id, name, author, pgenre\_id, id],(error, results) => {              if (error) {                  throw error              }              response.status(200).send(`Book modified with ID: ${id}`)          }          );  };  module.exports = {  getBook,  getBookById,  createBook,  **updateBook,**  deleteBook,  };  **Index.js**  app.put('/book/:id', db.updateBook); | PUT localhost:3000/book/1 |