```javascript
const library = [
{
title: "The Great Gatsby",
author: "F. Scott Fitzgerald",
year: 1925,
pages: 180,
},
{
title: "To Kill a Mockingbird",
author: "Harper Lee",
year: 1960,
pages: 281,
},
{
title: "1984",
author: "George Orwell",
year: 1949,
pages: 328,
},
{
title: "Pride and Prejudice",
author: "Jane Austen",
year: 1813,
pages: 432,
},
];
```

Total Number of Pages: Write a function getTotalPages that calculates and returns the total number of pages in the library.

```javascript
const getTotalPages=(library)=> {
const ans = library.reduce((total, book) => total + book.pages, 0);
return ans;
}
console.log(getTotalPages(library));
```

List of Book Titles: Write a function getBookTitles that returns an array containing only the titles of the books in the library.

```javascript
const getBookTitles=(library)=>{
return library.map((book) => book.title);
}
console.log(getBookTitles(library));
```

Books Published After a Given Year: Write a function getBooksPublishedAfterYear(year) that takes a year as a parameter and returns an array containing the titles of books published after that year.

```
const getBooksPublishedAfterYear=(library, year)=> {
const k= library.filter((book)=>book.year>year);
return k.map((book)=>book.title);
}
console.log(getBooksPublishedAfterYear(library,1800));
```

Average Number of Pages: Write a function getAveragePages that calculates and returns the average number of pages across all books in the library.

```
across all books in the library.
const getAveragePages=(library)=> {
return getTotalPages(library)/4;
}
console.log(getAveragePages(library));
```

Longest Book: Write a function getLongestBook that returns the title of the book with the most pages.

```
const getLongestBook=(library)=> {
const longestBook = library.reduce((maxBook, book) =>
book.pages > maxBook.pages ? book : maxBook
);
return longestBook.title;
}
console.log(getLongestBook(library));
```

Authors and Their Books: Write a function getAuthorsAndBooks that returns an object where the keys are author names and the values are arrays of book titles written by each author.

```
const getAuthorsAndBooks=(library) =>{
return library.reduce((authorsBooks, book) => {
if (!authorsBooks[book.author]) {
authorsBooks[book.author] = [];
}
authorsBooks[book.author].push(book.title);
return authorsBooks;
}, {});
}
```

```
console.log(getAuthorsAndBooks(library));
```

Total Number of Pages by Author: Write a function getTotalPagesByAuthor that returns an object where the keys are author names and the values are the total number of pages of books written by each author.

```
const getTotalPagesByAuthor=(library) =>{
return library.reduce((totalPagesByAuthor, book) => {
if (!totalPagesByAuthor[book.author]) {
totalPagesByAuthor[book.author] = 0;
}
totalPagesByAuthor[book.author] += book.pages;
return totalPagesByAuthor;
}, {});
}
console.log(getTotalPagesByAuthor(library));
```

Advanced: Filter and Map: Write a function getShortestBookByAuthor that returns an object where the keys are author names and the values are the titles of the shortest book written by each author.

```
const getShortestBookByAuthor=(library)=> {
return library.reduce((shortestBooksByAuthor, book) => {
if (!shortestBooksByAuthor[book.author]) {
shortestBooksByAuthor[book.author] = book.title;
} else if (book.pages < library.find((b) => b.title ===
shortestBooksByAuthor[book.author]).pages) {
shortestBooksByAuthor[book.author] = book.title;
}
return shortestBooksByAuthor;
}, {});

}
```