We have array of student object.

we have 2 example filter using age and birth_date

- In array we have name, class, roll, age in 1st example and birth_date in 2nd example.

Q1) age less than 10

- we use filter method as its main function is to filter data form given array.
- So we filter array according to age

const age10 = students.filter((student) => student.age < 10);

Q2) Calculate average age

```
const averageAge =
  students.reduce((acc, item) => {
    return acc + item.age;
  }, 0) / students.length;
console.log(averageAge);
```

we use reduce method

- It takes 2 value acc and item
- acc is accumulator, used to give initial value before calculation we assign it as argument in reduce function as 0
- each time reduce function perform calculation and store in result accumulator
- At last we get result of accumulator
- calculation is simple average calculation

Q3) Every child is gen-z or not

we use every function

- It return Boolean value
- if condition met return true else false
- So we check condition it return false and apply ternary operator to return No

Q4) list only genz student

we use filter with condition that satisfy gen-z.

```
function GenZOnly(students) {
  return students.filter((student) => student.age < 28 && student.age > 13);
}
```

Example 2

Using brith_date

- date is define by Date object
- Date.now() return current date in millisecond
- And getTime() function also return time in millisecond
- Time is calculated from 1970 January 1
- We subtract and convert it into years
- and add age property to object and filter it according to calculated age if it is genz or not.

```
let students1 = [
  { name: "Ram Parsad", class: 9, roll: 6, birth_date: new Date("2008-4-15")
},
  {
    name: "Hari Parsad",
    class: 5,
    roll: 43,
    birth_date: new Date("1987-2-16"),
  },
  {
    name: "Sita Parsad",
    class: 8,
    roll: 34,
    birth_date: new Date("2010-3-1"),
  },
  {
    name: "Gita Parsad",
    class: 10,
    roll: 6,
    birth_date: new Date("2008-6-15"),
  },
];
const d = Date.now();
console.log(d); // d is in millisecond
const newmap = students1.map((student) => {
  //getTime return time in millisecond
  const age = Math.floor(
    (d - student.birth_date.getTime()) / (1000 * 60 * 60 * 24 * 365) // to
millisecond in year
  );
  console.log(age);
  return { ...student, age }; //adding age property
});
console.log(newmap);
console.log(newmap.filter((student) => student.age < 28 && student.age >
13));
```