

```
export function sum(values) {
  let somme = 0;
  for (const value of values) {
    somme += value;
  }
  return somme;
}

export function addToAll(values, toAdd) {
  return values.map((value) => value + toAdd)
}

export function average(param) {
  return param.length === 0 ? null : sum(param) / param.length;
}

export function isAdult(person) {
  return person.age >= 18;
}

export function isChild(person) {
  return person.age < 18;
}

export function extractAge(users) {
  return users.map(user => user.age);
}

export function adultFilter(users, categorie) {
  if (categorie === "adult") {
    return users.filter(isAdult);
  }
  else if (categorie === "child") {
    return users.filter(isChild);
  }
  return users
}

export function ageAverage(persons, categorie) {
  const ages = extractAge(adultFilter(persons, categorie));
  return average(ages);
}

export function isAllAdult(persons) {
  for (const person of persons) {
    if (isChild(person))
      return false;
  }
  return true;
}

export function hasChild(persons) {
  return !isAllAdult(persons);
}

export function addIsAdultProperty(persons) {
  return persons.map((person) => {
    return {
      ...person,
      isAdult: isAdult(person),
    };
  });
}
```

})