

assets\js\script.js

```
/* **Not necessary and Not Finish (espère une "correction" ou conseils)** */

// Inputs radio Join , Contact , Donate
const inputRadios = document.querySelectorAll('input#join, input#contact, input#donate');

//Email & text inputs
const allInputs = document.querySelectorAll('input[type="text"], textarea');

//form
const form = document.querySelector('form');

let lastname, firstname, email, messageBox;

//fonction change attribute checked
const changeAttribute = () => {
  checkContent.addEventListener('click', (e) => {
    //Loop into the inputs radio
    for (let i = 0; i < inputRadios.length; i++) {
      if (!inputRadios[i].checked) {
        inputRadios[i].removeAttribute('checked', '');
        e.target.setAttribute('checked', 'true');
      }
    }
  });
};
changeAttribute();

//Function display form
const inputRadiosDisplay = () => {
  let textarea = document.querySelector('textarea');
  const contact = document.querySelector('input#contact');
  const join = document.querySelector('input#join');
  const donate = document.querySelector('input#donate');

  //
  contact.addEventListener('input', (e) => {
    if (e.target.id === 'contact' && e.target.checked === true) {
      changeAttribute();
      joinChecked.style.visibility = 'hidden';
      paymentRow.style.visibility = 'hidden';

      textarea.style.display = 'block';
    } else {
      textarea.style.display = 'none';
    }
  });
  join.addEventListener('input', (e) => {
    if (e.target.id === 'join' && e.target.checked === true) {
      changeAttribute();
      joinChecked.style.display = 'block';
      joinChecked.style.visibility = 'visible';
      paymentRow.style.visibility = 'hidden';
      paymentRow.style.display = 'none';
      textarea.style.display = 'none';
    } else {
      joinChecked.style.visibility = 'hidden';
    }
  });
};
```

```
        textarea.style.display = 'none';
    }
});
donate.addEventListener('input', (e) => {
    if (e.target.id === 'donate' && e.target.checked === true) {
        changeAttribute();
        paymentRow.style.visibility = 'visible';
        paymentRow.style.display = 'block';
        joinChecked.style.visibility = 'hidden';
        joinChecked.style.display = 'none';
        textarea.style.display = '';
    } else {
        textarea.style.display = 'block';
        paymentRow.style.visibility = 'hidden';
        paymentRow.style.display = '';
    }
});
});
inputRadiosDisplay();

//Functions display error message
const errorDisplay = (tag, message, valid) => {
    const containers = document.querySelector('.' + tag + '-container');
    const spans = document.querySelector('.' + tag + '-container > span');

    if (!valid) {
        containers.classList.add('error');
        spans.textContent = message;
    } else {
        containers.classList.remove('error');
        spans.textContent = message;
    }
};

//Function for checking lastname
const lastnameChecker = (value) => {
    if (value.length > 0 && (value.length < 2 || value.length > 20)) {
        errorDisplay('lastname', 'Veuillez saisir entre 2 et 20 caractères pour le champ du nom.');
        lastname = null;
    } else if (!value.match(/^[a-zA-Z]+$/)) {
        errorDisplay('lastname', 'Veuillez saisir uniquement des lettres pour le champ du nom.');
        lastname = null;
    } else {
        errorDisplay('lastname', '', true);
        lastname = value;
    }
};

//Function for checking firstname
const firstnameChecker = (value) => {
    if (value.length > 0 && (value.length < 2 || value.length > 20)) {
        errorDisplay('firstname', 'Veuillez saisir entre 2 et 20 caractères pour le champ du prénom.');
        firstname = null;
    } else if (!value.match(/^[a-zA-Z]+$/)) {
        errorDisplay('firstname', 'Veuillez saisir uniquement des lettres pour le champ du prénom.');
        firstname = null;
    } else {
        errorDisplay('firstname', '', true);
        firstname = value;
    }
};
```

```

    }
  };
  //Function for email address
  const emailChecker = (value) => {
    if (!value.match(/^[\w-\.\.]+\@([\w-]+\.\.)+[\w-]{2,4}$/i)) {
      errorDisplay('email', "L'e-mail n'est pas valide.");
      email = null;
    } else {
      errorDisplay('email', '', true);
      email = value;
    }
  };
  //Function for checking message area
  const messageAreaChecker = (value) => {
    if (value.length > 0 && value.length < 10) {
      errorDisplay('messageAera', 'Veuillez saisir au moins 10 caractères pour le champ du message.');
```

messageBox = null;

```

    } else {
      errorDisplay('messageAera', ' ', true);
      messageBox = value;
    }
  };
  //Function for checking values of all inputs
  allInputs.forEach((inputChoiced) => {
    inputChoiced.addEventListener('input', (e) => {
      // console.log(e.target.id); // Vérification de l'id de l'input
      switch (e.target.id) {
        case 'lastname':
          lastnameChecker(e.target.value);
          break;
        case 'firstname':
          firstnameChecker(e.target.value);
          break;
        case 'email':
          emailChecker(e.target.value);
          break;
        case 'messageArea':
          messageAreaChecker(e.target.value);
          break;

        default:
          null;
          break;
      }
    });
  });

  form.addEventListener('submit', (e) => {
    console.log(e.target.value);
    e.preventDefault();
    // If all inputs are valid then send data "to the server"
    if (lastname && firstname) {
      const data = {
        lastname,
        firstname,
        email,
        messageBox,
      };
      console.log(data);
    }
  });

```

```
// Reset all inputs
allInputs.forEach((input) => {
  input.value = '';
});

lastname = null;
firstname = null;
email = null;
messageBox = null;
alert("Formulaire validé pour l'exercice !");
} else {
  console.log('Tous les champs ne sont pas valides.');
```