JSS (script.jss):

```javascript
const calendarEl = document.getElementById("calendar");

const currentMonthEl = document.getElementById("current-month");

const prevMonthBtn = document.getElementById("prev-month");

const nextMonthBtn = document.getElementById("next-month");

const appointmentDateEl = document.getElementById("appointment-date");

const appointmentDetailsEl = document.getElementById("appointment-details");

const addAppointmentBtn = document.getElementById("add-appointment");

const appointmentListEl = document.getElementById("appointment-list");


let currentDate = new Date();
let appointments = [];



// Generate calendar for a specific month
function generateCalendar(date) {
    calendarEl.innerHTML = "";
    const firstDay = new Date(date.getFullYear(), date.getMonth(), 1);
    const lastDay = new Date(date.getFullYear(), date.getMonth() + 1, 0);
    const today = new Date();

    currentMonthEl.textContent = date.toLocaleDateString("en-US", {
        month: "long",
        year: "numeric",
```

```javascript
});

// Fill blank days before the month starts
for (let i = 0; i < firstDay.getDay(); i++) {
    const blank = document.createElement("div");
    calendarEl.appendChild(blank);
}

// Fill days of the month
for (let day = 1; day <= lastDay.getDate(); day++) {
    const dayEl = document.createElement("div");
    dayEl.textContent = day;
    dayEl.className = "day";

    const fullDate = `${date.getFullYear()}-${String(
        date.getMonth() + 1
    ).padStart(2, "0")}-${String(day).padStart(2, "0")}`;

    if (
        today.getFullYear() === date.getFullYear() &&
        today.getMonth() === date.getMonth() &&
        today.getDate() === day
    ) {
        dayEl.classList.add("today");
    }

    if (appointments.some((appt) => appt.date === fullDate)) {
```

```javascript
        dayEl.classList.add("appointment");
      }


      calendarEl.appendChild(dayEl);
    }
  }


// Add appointment
addAppointmentBtn.addEventListener("click", () => {
    const date = appointmentDateEl.value;
    const details = appointmentDetailsEl.value;


    if (date && details) {
        appointments.push({ date, details });
        updateAppointmentsList();
        generateCalendar(currentDate);
        appointmentDateEl.value = "";
        appointmentDetailsEl.value = "";
    } else {
        alert("Please fill in both fields.");
    }
});


// Update appointments list
function updateAppointmentsList() {
    appointmentListEl.innerHTML = "";
    appointments.forEach((appt) => {
```

```javascript
    const li = document.createElement("li");

    li.textContent = `${appt.date}: ${appt.details}`;

    appointmentListEl.appendChild(li);

  });

}


// Navigate months

prevMonthBtn.addEventListener("click", () => {

  currentDate.setMonth(currentDate.getMonth() - 1);

  generateCalendar(currentDate);

});


nextMonthBtn.addEventListener("click", () => {

  currentDate.setMonth(currentDate.getMonth() + 1);

  generateCalendar(currentDate);

});


// Initial load

generateCalendar(currentDate);
```