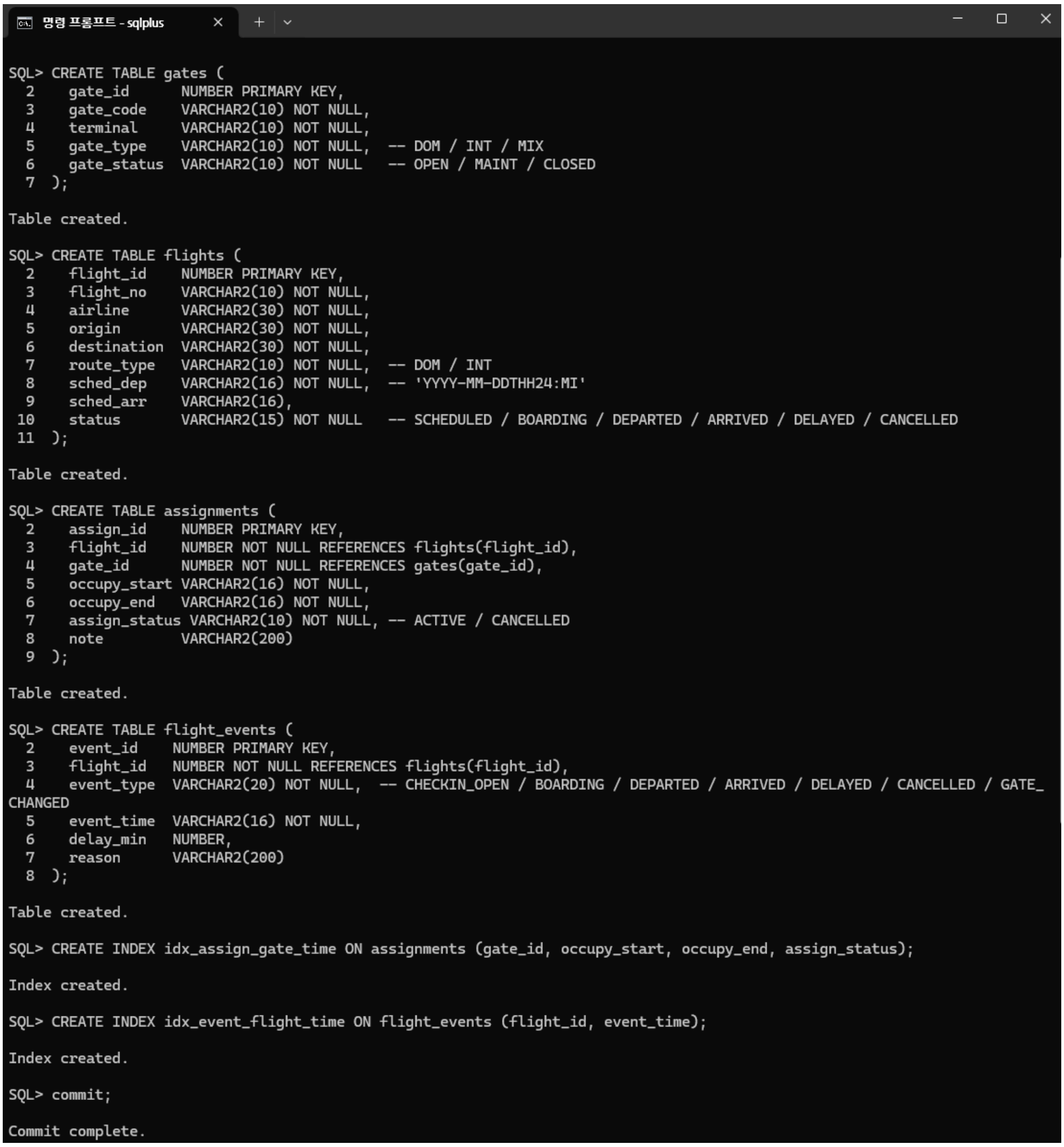
2025 빅데이터과 2학기 기말고사 과제 제출
학 번 : 2455025
학 과 : 빅데이터과
이 름 : 정재원
GITHUB :
https://github.com/GabrielJung0727/Airport-Flight-Gate-Operations-Management-System

1. SQLPLUS

```
sqlplus
/as sysdba

CREATE TABLE gates (
  gate_id      NUMBER PRIMARY KEY,
  gate_code    VARCHAR2(10) NOT NULL,
  terminal     VARCHAR2(10) NOT NULL,
  gate_type    VARCHAR2(10) NOT NULL,  -- DOM / INT / MIX
  gate_status  VARCHAR2(10) NOT NULL   -- OPEN / MAINT / CLOSED
);

CREATE TABLE flights (
  flight_id    NUMBER PRIMARY KEY,
  flight_no    VARCHAR2(10) NOT NULL,
  airline      VARCHAR2(30) NOT NULL,
  origin       VARCHAR2(30) NOT NULL,
  destination  VARCHAR2(30) NOT NULL,
  route_type   VARCHAR2(10) NOT NULL,  -- DOM / INT
  sched_dep    VARCHAR2(16) NOT NULL,  -- 'YYYY-MM-DDTHH24:MI'
  sched_arr    VARCHAR2(16),
  status       VARCHAR2(15) NOT NULL   -- SCHEDULED / BOARDING /
DEPARTED / ARRIVED / DELAYED / CANCELLED
);

CREATE TABLE assignments (
  assign_id    NUMBER PRIMARY KEY,
  flight_id    NUMBER NOT NULL REFERENCES flights(flight_id),
  gate_id      NUMBER NOT NULL REFERENCES gates(gate_id),
  occupy_start VARCHAR2(16) NOT NULL,
  occupy_end   VARCHAR2(16) NOT NULL,
  assign_status VARCHAR2(10) NOT NULL, -- ACTIVE / CANCELLED
  note         VARCHAR2(200)
);

CREATE TABLE flight_events (
  event_id    NUMBER PRIMARY KEY,
```

```sql
  flight_id    NUMBER NOT NULL REFERENCES flights(flight_id),
  event_type  VARCHAR2(20) NOT NULL,  -- CHECKIN_OPEN / BOARDING /
DEPARTED / ARRIVED / DELAYED / CANCELLED / GATE_CHANGED
  event_time  VARCHAR2(16) NOT NULL,
  delay_min   NUMBER,
  reason      VARCHAR2(200)
);

-- Helpful indexes
CREATE INDEX idx_assign_gate_time ON assignments (gate_id,
occupy_start, occupy_end, assign_status);
CREATE INDEX idx_event_flight_time ON flight_events (flight_id,
event_time);

commit;
```



```
SQL> CREATE TABLE gates (
  2    gate_id       NUMBER PRIMARY KEY,
  3    gate_code     VARCHAR2(10) NOT NULL,
  4    terminal      VARCHAR2(10) NOT NULL,
  5    gate_type     VARCHAR2(10) NOT NULL,   -- DOM / INT / MIX
  6    gate_status   VARCHAR2(10) NOT NULL    -- OPEN / MAINT / CLOSED
  7  );

Table created.

SQL> CREATE TABLE flights (
  2    flight_id     NUMBER PRIMARY KEY,
  3    flight_no     VARCHAR2(10) NOT NULL,
  4    airline       VARCHAR2(30) NOT NULL,
  5    origin        VARCHAR2(30) NOT NULL,
  6    destination   VARCHAR2(30) NOT NULL,
  7    route_type    VARCHAR2(10) NOT NULL,   -- DOM / INT
  8    sched_dep     VARCHAR2(16) NOT NULL,   -- 'YYYY-MM-DDTHH24:MI'
  9    sched_arr     VARCHAR2(16),
 10    status        VARCHAR2(15) NOT NULL    -- SCHEDULED / BOARDING / DEPARTED / ARRIVED / DELAYED / CANCELLED
 11  );

Table created.

SQL> CREATE TABLE assignments (
  2    assign_id     NUMBER PRIMARY KEY,
  3    flight_id     NUMBER NOT NULL REFERENCES flights(flight_id),
  4    gate_id       NUMBER NOT NULL REFERENCES gates(gate_id),
  5    occupy_start VARCHAR2(16) NOT NULL,
  6    occupy_end    VARCHAR2(16) NOT NULL,
  7    assign_status VARCHAR2(10) NOT NULL, -- ACTIVE / CANCELLED
  8    note          VARCHAR2(200)
  9  );

Table created.

SQL> CREATE TABLE flight_events (
  2    event_id     NUMBER PRIMARY KEY,
  3    flight_id    NUMBER NOT NULL REFERENCES flights(flight_id),
  4    event_type   VARCHAR2(20) NOT NULL,  -- CHECKIN_OPEN / BOARDING / DEPARTED / ARRIVED / DELAYED / CANCELLED / GATE_
CHANGED
  5    event_time   VARCHAR2(16) NOT NULL,
  6    delay_min    NUMBER,
  7    reason       VARCHAR2(200)
  8  );

Table created.

SQL> CREATE INDEX idx_assign_gate_time ON assignments (gate_id, occupy_start, occupy_end, assign_status);

Index created.

SQL> CREATE INDEX idx_event_flight_time ON flight_events (flight_id, event_time);

Index created.

SQL> commit;

Commit complete.
```

2. app.css

```
:root{
  --bg: #0b1220;
  --panel: rgba(255,255,255,.06);
  --panel2: rgba(255,255,255,.08);
  --text: rgba(255,255,255,.92);
  --muted: rgba(255,255,255,.68);
  --border: rgba(255,255,255,.12);
  --brand: #7c3aed;
  --brand2:#22c55e;
  --danger:#ef4444;
  --warn:#f59e0b;
  --shadow: 0 16px 60px rgba(0,0,0,.35);
  --radius: 16px;
```

```css
  --radius2: 12px;
  --font: ui-sans-serif, system-ui, -apple-system, "Segoe UI", Roboto, "Noto Sans KR", Arial, "Apple SD Gothic Neo";
}

[data-theme="light"]{
  --bg: #f6f7fb;
  --panel: rgba(10,13,20,.06);
  --panel2: rgba(10,13,20,.08);
  --text: rgba(10,13,20,.92);
  --muted: rgba(10,13,20,.65);
  --border: rgba(10,13,20,.14);
  --shadow: 0 16px 60px rgba(10,13,20,.12);
}

*{ box-sizing:border-box; }
html,body{ height:100%; }
body{
  margin:0;
  font-family: var(--font);
  color: var(--text);
  background:
    radial-gradient(1200px 800px at 20% 10%, rgba(124,58,237,.28), transparent 60%),
    radial-gradient(900px 600px at 85% 20%, rgba(34,197,94,.22), transparent 60%),
    radial-gradient(900px 900px at 35% 90%, rgba(59,130,246,.16), transparent 60%),
    var(--bg);
}

a{ color:inherit; text-decoration:none; }
a:hover{ text-decoration:underline; }

.topbar{
  position: sticky;
  top: 0;
  z-index: 50;
  backdrop-filter: blur(14px);
  background: linear-gradient(to bottom, rgba(0,0,0,.35), rgba(0,0,0,.12));
  border-bottom: 1px solid var(--border);
}
```

```
[data-theme="light"] .topbar{
  background: linear-gradient(to bottom, rgba(255,255,255,.85),
rgba(255,255,255,.60));
}

.topbar-inner{
  max-width: 1120px;
  margin: 0 auto;
  padding: 14px 18px;
  display:flex;
  align-items:center;
  justify-content:space-between;
  gap:16px;
}

.brand{
  display:flex;
  align-items:center;
  gap:12px;
  font-weight: 800;
  letter-spacing: .2px;
}
.logo{
  width: 36px; height: 36px;
  border-radius: 12px;
  background: linear-gradient(135deg, var(--brand), #06b6d4);
  box-shadow: 0 10px 30px rgba(124,58,237,.35);
}

.nav{
  display:flex;
  flex-wrap:wrap;
  gap:10px;
  align-items:center;
}

.chip{
  padding: 8px 10px;
  border: 1px solid var(--border);
  background: var(--panel);
  border-radius: 999px;
  color: var(--muted);
  font-weight: 650;
  font-size: 13px;
  line-height: 1;
  transition: transform .15s ease, background .15s ease, color .15s
ease;
}
.chip:hover{
  transform: translateY(-1px);
  background: var(--panel2);
  color: var(--text);
  text-decoration:none;
}
.chip[aria-current="page"]{
  color: var(--text);
  border-color: rgba(124,58,237,.55);
  background: linear-gradient(135deg, rgba(124,58,237,.22),
rgba(34,197,94,.14));
}

.actions{ display:flex; gap:10px; align-items:center; }
.btn{
  appearance:none;
  border: 1px solid var(--border);
  background: var(--panel);
  color: var(--text);
  padding: 9px 12px;
  border-radius: 12px;
  font-weight: 700;
  font-size: 13px;
  cursor:pointer;
  transition: transform .15s ease, background .15s ease, border-color
.15s ease;
}
.btn:hover{ transform: translateY(-1px); background: var(--panel2); }
.btn.primary{
  border-color: rgba(124,58,237,.55);
  background: linear-gradient(135deg, rgba(124,58,237,.26),
rgba(124,58,237,.12));
}
.btn.danger{
  border-color: rgba(239,68,68,.55);
  background: linear-gradient(135deg, rgba(239,68,68,.22),
rgba(239,68,68,.10));
}
```

```
.container{
  max-width: 1120px;
  margin: 0 auto;
  padding: 22px 18px 56px;
}

.hero{
  border: 1px solid var(--border);
  background: linear-gradient(135deg, rgba(255,255,255,.08),
rgba(255,255,255,.04));
  border-radius: var(--radius);
  padding: 18px;
  box-shadow: var(--shadow);
}
.hero h1{
  margin: 0 0 6px;
  font-size: 22px;
  letter-spacing: -.2px;
}
.hero p{
  margin:0;
  color: var(--muted);
  line-height: 1.5;
}

.grid{
  margin-top: 16px;
  display:grid;
  grid-template-columns: repeat(12, 1fr);
  gap: 14px;
}
.card{
  grid-column: span 12;
  border: 1px solid var(--border);
  background: var(--panel);
  border-radius: var(--radius);
  box-shadow: var(--shadow);
  overflow:hidden;
}
.card-header{
  padding: 14px 16px;
  display:flex;
  align-items:center;
  justify-content:space-between;
  gap: 12px;
  border-bottom: 1px solid var(--border);
  background: linear-gradient(to bottom, rgba(255,255,255,.06),
transparent);
}
.card-title{
  margin:0;
  font-weight: 800;
  font-size: 16px;
}
.card-sub{
  margin: 2px 0 0;
  font-size: 13px;
  color: var(--muted);
}
.card-body{ padding: 16px; }

.toolbar{
  display:flex;
  gap:10px;
  flex-wrap:wrap;
  align-items:center;
  justify-content:space-between;
  margin-bottom: 10px;
}
.input, select, textarea{
  width: 100%;
  padding: 10px 12px;
  border-radius: 12px;
  border: 1px solid var(--border);
  background: rgba(255,255,255,.06);
  color: var(--text);
  outline: none;
}
[data-theme="light"] .input,
[data-theme="light"] textarea{
  background: rgba(10,13,20,.04);
}
select{
  cursor:pointer;
  appearance:none;
  -webkit-appearance:none;
  -moz-appearance:none;
  background: rgba(255,255,255,.08);
  color: var(--text);
  padding-right: 36px;
  background-image:
    linear-gradient(45deg, transparent 50%, var(--text) 50%),
    linear-gradient(135deg, var(--text) 50%, transparent 50%),
    linear-gradient(to right, transparent, transparent);
  background-position:
    calc(100% - 22px) calc(50% - 3px),
    calc(100% - 16px) calc(50% - 3px),
    calc(100% - 2.3em) 0;
  background-size: 6px 6px, 6px 6px, 1px 100%;
  background-repeat: no-repeat;
}
[data-theme="light"] select{
  background: rgba(10,13,20,.02);
}
select option{
  background-color: #0f172a;
  color: #f8fafc;
}
[data-theme="light"] select option{
  background-color: #f8fafc;
  color: #0f172a;
}
textarea{ resize: vertical; min-height: 80px; }
.input:focus, select:focus, textarea:focus{
  border-color: rgba(124,58,237,.65);
  box-shadow: 0 0 0 4px rgba(124,58,237,.18);
}
.field{
  display:flex;
  flex-direction:column;
  gap: 6px;
}
.label{
  font-size: 12px;
  color: var(--muted);
  font-weight: 750;
}
.row{
  display:grid;
  grid-template-columns: repeat(12, 1fr);
  gap: 12px;
}
.col-6{ grid-column: span 6; }
.col-4{ grid-column: span 4; }
.col-3{ grid-column: span 3; }
.col-12{ grid-column: span 12; }

@media (max-width: 860px){
  .col-6,.col-4,.col-3{ grid-column: span 12; }
  .nav{ display:none; }
}

.table-wrap{
  border: 1px solid var(--border);
  border-radius: var(--radius2);
  overflow:auto;
}
table{
  width: 100%;
  border-collapse: collapse;
  min-width: 780px;
}
th, td{
  padding: 11px 12px;
  border-bottom: 1px solid var(--border);
  font-size: 13px;
}
th{
  text-align:left;
  color: var(--muted);
  font-weight: 800;
  background: rgba(255,255,255,.04);
  position: sticky;
  top: 0;
  z-index: 1;
}
tr:hover td{ background: rgba(255,255,255,.04); }

.badge{
  display:inline-flex;
  align-items:center;
  gap:6px;
  padding: 4px 10px;
  border-radius: 999px;
  border: 1px solid var(--border);
  background: rgba(255,255,255,.06);
  font-weight: 800;
  font-size: 12px;
  color: var(--muted);
}
.badge.ok{ color: rgba(34,197,94,.95); border-color:
rgba(34,197,94,.35); }
.badge.warn{ color: rgba(245,158,11,.95); border-color:
rgba(245,158,11,.35); }
.badge.bad{ color: rgba(239,68,68,.95); border-color:
rgba(239,68,68,.35); }
.badge.purple{ color: rgba(167,139,250,.98); border-color:
rgba(167,139,250,.35); }

.hint{ color: var(--muted); font-size: 13px; line-height: 1.5; }
.split{
  display:flex;
  gap:10px;
  flex-wrap:wrap;
}
.w-260{ width: 260px; max-width: 100%; }
.w-360{ width: 360px; max-width: 100%; }

.toast{
  position: fixed;
  right: 18px;
  bottom: 18px;
  z-index: 80;
  width: min(420px, calc(100vw - 36px));
  border: 1px solid var(--border);
  background: rgba(20,25,38,.86);
  backdrop-filter: blur(14px);
  border-radius: 16px;
  box-shadow: var(--shadow);
  padding: 12px 12px;
  display:none;
}
[data-theme="light"] .toast{ background: rgba(255,255,255,.86); }
.toast.show{ display:block; animation: pop .16s ease-out; }
```

```
@keyframes pop{ from{ transform: translateY(6px); opacity:.0 } to{
transform: translateY(0); opacity:1 } }
.toast-title{ font-weight: 900; margin:0 0 2px; font-size: 14px; }
.toast-msg{ margin:0; color: var(--muted); font-size: 13px;
line-height: 1.4; }
.toast.success{ border-color: rgba(34,197,94,.35); }
.toast.error{ border-color: rgba(239,68,68,.35); }
.toast.warn{ border-color: rgba(245,158,11,.35); }

.footer{
  margin-top: 28px;
  color: var(--muted);
  font-size: 12px;
  text-align:center;
}

.guide-panel{
  margin-top: 12px;
  border: 1px solid var(--border);
  border-radius: var(--radius2);
  background: rgba(255,255,255,.04);
  padding: 14px;
  display:none;
}
[data-theme="light"] .guide-panel{
  background: rgba(10,13,20,.04);
}
.guide-panel.show{
  display:block;
  animation: pop .16s ease-out;
}
.guide-panel h4{
  margin:0 0 8px;
  font-size: 14px;
}
.guide-panel ul{
  margin:0 0 8px 18px;
  padding:0;
  color: var(--muted);
  font-size: 13px;
  line-height: 1.4;
}
.guide-panel a{
  color: var(--brand2);
  font-weight: 700;
}
```

3. app.js

```
(function () {
  const root = document.documentElement;

  function setTheme(theme) {
    root.setAttribute("data-theme", theme);
    try { localStorage.setItem("theme", theme); } catch (_) {}
  }

  function initTheme() {
    const saved = (() => {
      try { return localStorage.getItem("theme"); } catch (_) { return
null; }
    })();
    if (saved === "light" || saved === "dark") return setTheme(saved);
    setTheme("dark");
  }

  function qs(sel, el) { return (el || document).querySelector(sel); }
  function qsa(sel, el) { return Array.from((el ||
document).querySelectorAll(sel)); }

  function initThemeToggle() {
    const btn = qs("[data-theme-toggle]");
    if (!btn) return;
    btn.addEventListener("click", () => {
      const current = root.getAttribute("data-theme") || "dark";
      setTheme(current === "dark" ? "light" : "dark");
    });
  }

  function setActiveNav() {
    const path = (location.pathname || "").split("/").pop();
    qsa(".chip[data-page]").forEach((a) => {
      if (a.getAttribute("data-page") === path)
a.setAttribute("aria-current", "page");
    });
  }

  function initTableFilter() {
    qsa("[data-table-filter]").forEach((input) => {
      const tableId = input.getAttribute("data-table-filter");
      const table = qs("#" + tableId);
      if (!table) return;
      const rows = qsa("tbody tr", table);
      input.addEventListener("input", () => {
        const q = (input.value || "").toLowerCase().trim();
        rows.forEach((tr) => {
          const text = (tr.innerText || "").toLowerCase();
          tr.style.display = text.includes(q) ? "" : "none";
        });
      });
    });
  }

  function toast(type, title, msg) {
    const t = qs("#toast");
    if (!t) return;
    t.className = "toast " + (type || "");
    qs(".toast-title", t).textContent = title || "알림";
    qs(".toast-msg", t).textContent = msg || "";
    t.classList.add("show");
    setTimeout(() => t.classList.remove("show"), 3200);
  }

  function initToastFromQuery() {
    const params = new URLSearchParams(location.search);
    const msg = params.get("msg");
    if (!msg) return;
    const type = params.get("type") || "success";
    const title = params.get("title") || (type === "error" ? "오류" :
"완료");
    toast(type, title, msg);
  }

  function initAutoOccupyTime() {
    const flightSel = qs("[data-flight-select]");
    const startEl = qs("[data-occupy-start]");
    const endEl = qs("[data-occupy-end]");
    const btn = qs("[data-occupy-auto]");
    if (!flightSel || !startEl || !endEl) return;
    function parseLocalDT(dt) {
      if (!dt || dt.length < 16) return null;
      const y = +dt.slice(0, 4);
      const m = +dt.slice(5, 7) - 1;
      const d = +dt.slice(8, 10);
      const hh = +dt.slice(11, 13);
      const mm = +dt.slice(14, 16);
      const date = new Date(y, m, d, hh, mm, 0, 0);
      return Number.isNaN(date.getTime()) ? null : date;
    }

    function fmtLocalDT(date) {
      const pad = (n) => String(n).padStart(2, "0");
      return (
        date.getFullYear() +
        "-" + pad(date.getMonth() + 1) +
        "-" + pad(date.getDate()) +
        "T" + pad(date.getHours()) +
        ":" + pad(date.getMinutes())
      );
    }

    function applyAuto() {
      const opt = flightSel.options[flightSel.selectedIndex];
      const dep = opt ? opt.getAttribute("data-dep") : null;
      const depDate = parseLocalDT(dep);
      if (!depDate) return toast("warn", "시간 자동계산", "선택한 항공편의 예정 출발시간이 올바르지 않습니다.");
      const start = new Date(depDate.getTime() - 60 * 60000);
      const end = new Date(depDate.getTime() + 15 * 60000);
      startEl.value = fmtLocalDT(start);
      endEl.value = fmtLocalDT(end);
      toast("success", "시간 자동계산", "점유 시간(출발 60분 전 ~ 출발 15분 후)으로 채웠습니다.");
    }

    if (btn) btn.addEventListener("click", (e) => { e.preventDefault(); applyAuto(); });
    flightSel.addEventListener("change", () => {
      if (startEl.value || endEl.value) return;
      applyAuto();
    });
  }
  initTheme();
  initThemeToggle();
  setActiveNav();
  initTableFilter();
  initToastFromQuery();
  initAutoOccupyTime();
  initGateGuideToggle();

  function initGateGuideToggle() {
    qsa("[data-gate-guide-toggle]").forEach((btn) => {
      const target = btn.getAttribute("data-gate-guide-toggle");
      const panel = target ? qs(target) : null;
      if (!panel) return;
      btn.addEventListener("click", () => {
        panel.classList.toggle("show");
      });
    });
  }
})();
```

4. assign_action.jsp

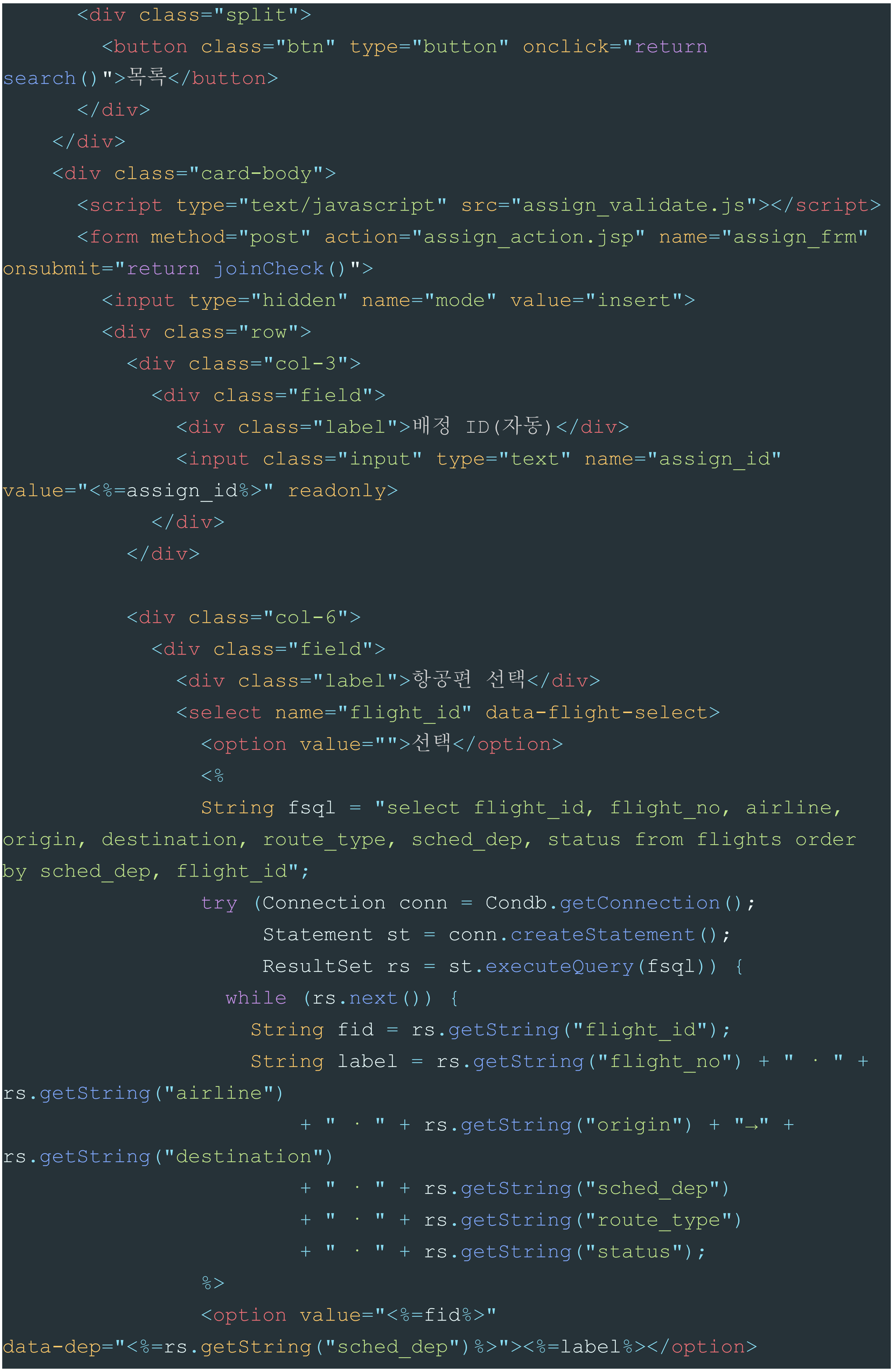
```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");

String mode = request.getParameter("mode");
String assign_id = request.getParameter("assign_id");
String flight_id = request.getParameter("flight_id");
String gate_id = request.getParameter("gate_id");
String occupy_start = request.getParameter("occupy_start");
String occupy_end = request.getParameter("occupy_end");
String assign_status = request.getParameter("assign_status");
String note = request.getParameter("note");

if (mode == null) {

response.sendRedirect("assign_list.jsp?type=error&msg=mode%20%EB%88%84%
EB%9D%BD");
  return;
```

```
}

boolean shouldCheck = "ACTIVE".equals(assign_status);

String returnPage = "insert".equals(mode) ? "assign_add.jsp" :
"assign_edit.jsp?assign_id=" + assign_id;
String returnSep = returnPage.contains("?") ? "&" : "?";

try (Connection conn = Condb.getConnection()) {
  if (shouldCheck) {
    // Gate status check (MAINT/CLOSED block)
    try (PreparedStatement ps = conn.prepareStatement("select
gate_status, gate_type from gates where gate_id=?")) {
      ps.setInt(1, Integer.parseInt(gate_id));
      try (ResultSet rs = ps.executeQuery()) {
        if (rs.next()) {
          String gateStatus = rs.getString("gate_status");
          if (!"OPEN".equals(gateStatus)) {
            response.sendRedirect(returnPage + returnSep +
"type=error&msg=%EA%B2%8C%EC%9D%B4%ED%8A%B8%20%EC%83%81%ED%83%9C%EA%B0%
80%20OPEN%EC%9D%B4%20%EC%95%84%EB%8B%99%EB%8B%88%EB%8B%A4");
            return;
          }
        }
      }
    }

    // Route type vs gate type check
    String routeType = null;
    String gateType = null;
    try (PreparedStatement ps = conn.prepareStatement(
      "select f.route_type, g.gate_type from flights f join gates g on
g.gate_id=? where f.flight_id=?")) {
      ps.setInt(1, Integer.parseInt(gate_id));
      ps.setInt(2, Integer.parseInt(flight_id));
      try (ResultSet rs = ps.executeQuery()) {
        if (rs.next()) {
          routeType = rs.getString(1);
          gateType = rs.getString(2);
        }
      }
    }
    if (routeType != null && gateType != null) {
      if ("INT".equals(routeType) && "DOM".equals(gateType)) {
        response.sendRedirect(returnPage + returnSep +
"type=error&msg=%EA%B5%AD%EC%A0%9C%ED%8E%B8%EC%9D%80%20DOM%20%EA%B2%8C%EC%9D%B4%ED%8A%B8%EC%97%90%20%EB%B0%B0%EC%A0%95%ED%95%A0%20%EC%88%98%20%EC%97%86%EC%8A%B5%EB%8B%88%EB%8B%A4");
        return;
      }
      if ("DOM".equals(routeType) && "INT".equals(gateType)) {
        response.sendRedirect(returnPage + returnSep +
"type=error&msg=%EA%B5%AD%EB%82%B4%ED%8E%B8%EC%9D%80%20INT%20%EA%B2%8C%EC%9D%B4%ED%8A%B8%EC%97%90%20%EB%B0%B0%EC%A0%95%ED%95%A0%20%EC%88%98%20%EC%97%86%EC%8A%B5%EB%8B%88%EB%8B%A4");
        return;
      }
    }

    // Overlap check (same gate, ACTIVE, intersect time)
    String overlapSql =
      "select count(*) cnt from assignments " +
      "where gate_id=? and assign_status='ACTIVE' " +
      "and occupy_start < ? and occupy_end > ? " +
      (("modify".equals(mode)) ? "and assign_id <> ? " : "");
    try (PreparedStatement ps = conn.prepareStatement(overlapSql)) {
      ps.setInt(1, Integer.parseInt(gate_id));
      ps.setString(2, occupy_end);
      ps.setString(3, occupy_start);
      if ("modify".equals(mode)) {
        ps.setInt(4, Integer.parseInt(assign_id));
      }
      try (ResultSet rs = ps.executeQuery()) {
        rs.next();
        if (rs.getInt("cnt") > 0) {
          response.sendRedirect(returnPage + returnSep +
"type=error&msg=%EA%B2%8C%EC%9D%B4%ED%8A%B8%20%EC%A4%91%EB%B3%B5%20%EB%B0%B0%EC%A0%95(%EC%8B%9C%EA%B0%84%20%EA%B2%B9%EC%B9%A8)%EC%9E%85%EB%8B%88%EB%8B%A4");
          return;
        }
      }
    }
  }

  if ("insert".equals(mode)) {
    String sql = "insert into assignments (assign_id, flight_id,
gate_id, occupy_start, occupy_end, assign_status, note) values
(?,?,?,?,?,?,?)";
    try (PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setInt(1, Integer.parseInt(assign_id));
      ps.setInt(2, Integer.parseInt(flight_id));
      ps.setInt(3, Integer.parseInt(gate_id));
      ps.setString(4, occupy_start);
      ps.setString(5, occupy_end);
      ps.setString(6, assign_status);
      ps.setString(7, (note == null || note.trim().isEmpty()) ? null :
note);
      ps.executeUpdate();
    }

response.sendRedirect("assign_add.jsp?type=success&msg=%EA%B2%8C%EC%9D%
B4%ED%8A%B8%20%EB%B0%B0%EC%A0%95%20%EB%93%B1%EB%A1%9D%20%EC%99%84%EB%A3
%8C");
    return;
  }

  if ("modify".equals(mode)) {
    String sql = "update assignments set gate_id=?, occupy_start=?,
occupy_end=?, assign_status=?, note=? where assign_id=?";
    try (PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setInt(1, Integer.parseInt(gate_id));
      ps.setString(2, occupy_start);
      ps.setString(3, occupy_end);
      ps.setString(4, assign_status);
      ps.setString(5, (note == null || note.trim().isEmpty()) ? null :
note);
      ps.setInt(6, Integer.parseInt(assign_id));
      ps.executeUpdate();
    }

response.sendRedirect("assign_list.jsp?type=success&msg=%EB%B0%B0%EC%A0
%95%20%EC%88%98%EC%A0%95%20%EC%99%84%EB%A3%8C");
    return;
  }
} catch (Exception e) {
  e.printStackTrace();
```

```
response.sendRedirect("assign_list.jsp?type=error&msg=%EC%B2%98%EB%A6%A
C%20%EC%A4%91%20%EC%98%A4%EB%A5%98%20%EB%B0%9C%EC%83%9D");
  return;
}

response.sendRedirect("assign_list.jsp?type=error&msg=%EC%95%8C%20%EC%8
8%98%20%EC%97%86%EB%8A%94%20mode");
%>
```

5. assign_add.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");
String assign_id = "";
try (Connection conn = Condb.getConnection();
     Statement st = conn.createStatement();
     ResultSet rs = st.executeQuery("select nvl(max(assign_id),0)+1
assign_id from assignments")) {
  rs.next();
  assign_id = rs.getString("assign_id");
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>게이트 배정 등록</h1>
  <p>게이트 중복 배정(시간 구간 겹침)은 자동으로 차단됩니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">배정 정보 입력</p>
        <p class="card-sub">"시간 자동계산"을 사용하면 출발 기준으로 점유
시간이 채워집니다.</p>
      </div>
```

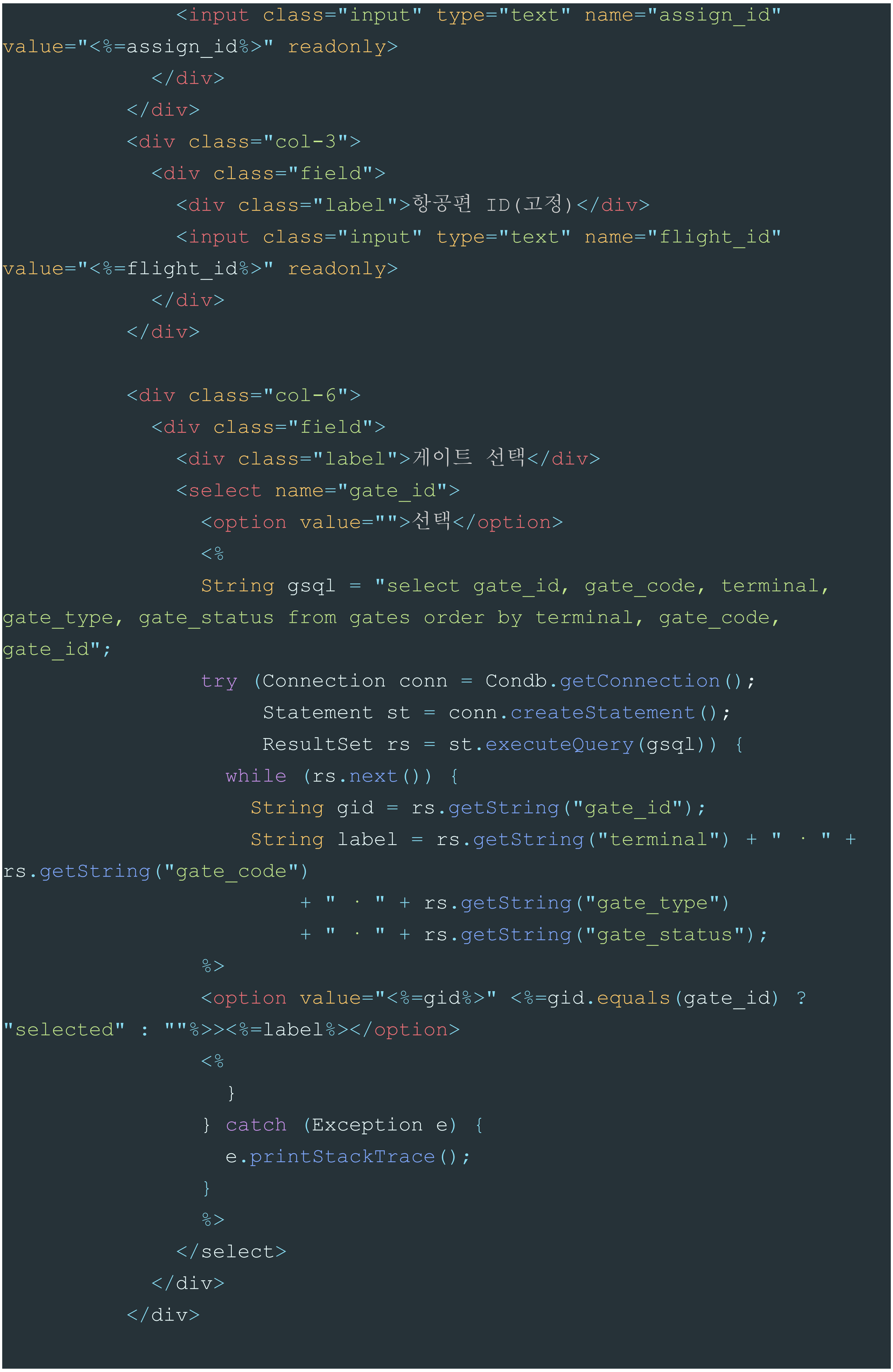


```
      <div class="split">
        <button class="btn" type="button" onclick="return
search()">목록</button>
      </div>
    </div>
    <div class="card-body">
      <script type="text/javascript" src="assign_validate.js"></script>
      <form method="post" action="assign_action.jsp" name="assign_frm"
onsubmit="return joinCheck()">
        <input type="hidden" name="mode" value="insert">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">배정 ID(자동)</div>
              <input class="input" type="text" name="assign_id"
value="<%=assign_id%>" readonly>
            </div>
          </div>

          <div class="col-6">
            <div class="field">
              <div class="label">항공편 선택</div>
              <select name="flight_id" data-flight-select>
                <option value="">선택</option>
                <%
                String fsql = "select flight_id, flight_no, airline,
origin, destination, route_type, sched_dep, status from flights order
by sched_dep, flight_id";
                try (Connection conn = Condb.getConnection();
                     Statement st = conn.createStatement();
                     ResultSet rs = st.executeQuery(fsql)) {
                  while (rs.next()) {
                    String fid = rs.getString("flight_id");
                    String label = rs.getString("flight_no") + " · " +
rs.getString("airline")
                        + " · " + rs.getString("origin") + "→" +
rs.getString("destination")
                        + " · " + rs.getString("sched_dep")
                        + " · " + rs.getString("route_type")
                        + " · " + rs.getString("status");
                %>
                <option value="<%=fid%>"
data-dep="<%=rs.getString("sched_dep")%>"><%=label%></option>
```

```
                <%
                  }
                } catch (Exception e) {
                  e.printStackTrace();
                }
                %>
              </select>
            </div>
          </div>

          <div class="col-6">
            <div class="field">
              <div class="label">게이트 선택</div>
              <select name="gate_id">
                <option value="">선택</option>
                <%
                String gsql = "select gate_id, gate_code, terminal, gate_type, gate_status from gates order by terminal, gate_code, gate_id";
                try (Connection conn = Condb.getConnection();
                     Statement st = conn.createStatement();
                     ResultSet rs = st.executeQuery(gsql)) {
                  while (rs.next()) {
                    String gid = rs.getString("gate_id");
                    String label = rs.getString("terminal") + " · " + rs.getString("gate_code")
                        + " · " + rs.getString("gate_type")
                        + " · " + rs.getString("gate_status");
                %>
                <option value="<%=gid%>"><%=label%></option>
                <%
                  }
                } catch (Exception e) {
                  e.printStackTrace();
                }
                %>
              </select>
            </div>
          </div>

          <div class="col-3">
            <div class="field">
              <div class="label">점유 시작</div>
              <input class="input" type="datetime-local"
name="occupy_start" data-occupy-start>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">점유 종료</div>
              <input class="input" type="datetime-local"
name="occupy_end" data-occupy-end>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">배정 상태</div>
              <select name="assign_status">
                <option value="ACTIVE" selected>ACTIVE</option>
                <option value="CANCELLED">CANCELLED</option>
              </select>
            </div>
          </div>
          <div class="col-12">
            <div class="field">
              <div class="label">비고(선택)</div>
              <textarea name="note" placeholder="예: 항공기 교체 예정,
승무원 교대 등"></textarea>
            </div>
          </div>

          <div class="col-12">
            <div class="split">
              <button class="btn" type="button" data-occupy-auto>시간
자동계산</button>
              <button class="btn primary" type="submit">등록</button>
              <button class="btn" type="reset">초기화</button>
            </div>
          </div>
        </div>
      </form>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

6. assign_edit.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");

String assign_id = request.getParameter("assign_id");

String flight_id = "";
String flight_no = "";
String airline = "";
String origin = "";
String destination = "";
String sched_dep = "";
String route_type = "";

String gate_id = "";
String occupy_start = "";
String occupy_end = "";
String assign_status = "";
String note = "";

String sql =
  "select a.flight_id, a.gate_id, a.occupy_start, a.occupy_end,
a.assign_status, nvl(a.note,'') note, " +
  "f.flight_no, f.airline, f.origin, f.destination, f.sched_dep,
f.route_type " +
  "from assignments a join flights f on f.flight_id=a.flight_id where
a.assign_id=?";

try (Connection conn = Condb.getConnection();
     PreparedStatement ps = conn.prepareStatement(sql)) {
  ps.setInt(1, Integer.parseInt(assign_id));
  try (ResultSet rs = ps.executeQuery()) {
    if (rs.next()) {
      flight_id = rs.getString("flight_id");
      gate_id = rs.getString("gate_id");
      occupy_start = rs.getString("occupy_start");
      occupy_end = rs.getString("occupy_end");
      assign_status = rs.getString("assign_status");
      note = rs.getString("note");

      flight_no = rs.getString("flight_no");
      airline = rs.getString("airline");
      origin = rs.getString("origin");
      destination = rs.getString("destination");
      sched_dep = rs.getString("sched_dep");
      route_type = rs.getString("route_type");
    }
  }
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>배정 수정/취소</h1>
  <p>게이트/시간/상태를 변경합니다. ACTIVE 상태는 중복 배정 검사를 통과해야 합니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">배정 상세</p>
        <p class="card-sub"><%=flight_no%> · <%=airline%> · <%=origin%>→<%=destination%> · <%=sched_dep%> · <%=route_type%></p>
      </div>
      <div class="split">
        <a class="btn" href="assign_list.jsp">목록</a>
      </div>
    </div>
    <div class="card-body">
      <script type="text/javascript" src="assign_validate.js"></script>
      <form method="post" action="assign_action.jsp" name="assign_frm" onsubmit="return joinCheck()">
        <input type="hidden" name="mode" value="modify">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">배정 ID</div>
```



```
              <input class="input" type="text" name="assign_id"
value="<%=assign_id%>" readonly>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">항공편 ID(고정)</div>
              <input class="input" type="text" name="flight_id"
value="<%=flight_id%>" readonly>
            </div>
          </div>

          <div class="col-6">
            <div class="field">
              <div class="label">게이트 선택</div>
              <select name="gate_id">
                <option value="">선택</option>
                <%
                String gsql = "select gate_id, gate_code, terminal,
gate_type, gate_status from gates order by terminal, gate_code,
gate_id";
                try (Connection conn = Condb.getConnection();
                     Statement st = conn.createStatement();
                     ResultSet rs = st.executeQuery(gsql)) {
                  while (rs.next()) {
                    String gid = rs.getString("gate_id");
                    String label = rs.getString("terminal") + " · " +
rs.getString("gate_code")
                        + " · " + rs.getString("gate_type")
                        + " · " + rs.getString("gate_status");
                %>
                <option value="<%=gid%>" <%=gid.equals(gate_id) ?
"selected" : ""%>><%=label%></option>
                <%
                  }
                } catch (Exception e) {
                  e.printStackTrace();
                }
                %>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">점유 시작</div>
              <input class="input" type="datetime-local"
name="occupy_start" value="<%=occupy_start%>" data-occupy-start>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">점유 종료</div>
              <input class="input" type="datetime-local"
name="occupy_end" value="<%=occupy_end%>" data-occupy-end>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">배정 상태</div>
              <select name="assign_status">
                <option value="ACTIVE" <%=
"ACTIVE".equals(assign_status) ? "selected" : "" %>>ACTIVE</option>
                <option value="CANCELLED" <%=
"CANCELLED".equals(assign_status) ? "selected" : ""
%>>CANCELLED</option>
              </select>
            </div>
          </div>

          <div class="col-12">
            <div class="field">
              <div class="label">비고</div>
              <textarea name="note"><%=note%></textarea>
            </div>
          </div>

          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit" onclick="return
modify()">수정</button>
              <a class="btn" href="assign_list.jsp">취소</a>
            </div>
          </div>
        </div>
      </form>
```

```
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

7. assign_list.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>배정 조회/수정</h1>
  <p>항공편과 게이트 배정 내역을 확인하고, 배정 ID를 클릭하여 변경/취소할 수
있습니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">배정 목록</p>
        <p class="card-sub">ACTIVE 배정은 게이트 중복 배정 방지 규칙의
대상입니다.</p>
      </div>
      <div class="split">
        <a class="btn primary" href="assign_add.jsp">배정 등록</a>
      </div>
    </div>
    <div class="card-body">
      <div class="toolbar">
        <div class="split">
          <input class="input w-360" type="text" placeholder="검색:
편명/게이트/시간/상태" data-table-filter="assignTable">
        </div>
        <div class="hint">수정하려면 배정 ID를 클릭</div>
      </div>

      <div class="table-wrap">
        <table id="assignTable">
          <thead>
```

```
            <tr>
              <th>배정 ID</th>
              <th>항공편</th>
              <th>게이트</th>
              <th>점유 시작</th>
              <th>점유 종료</th>
              <th>상태</th>
              <th>비고</th>
            </tr>
          </thead>
          <tbody>
          <%
          request.setCharacterEncoding("UTF-8");
          String sql =
            "select a.assign_id, a.occupy_start, a.occupy_end, a.assign_status, nvl(a.note,'') note, " +
            "f.flight_no, f.airline, f.origin, f.destination, g.terminal, g.gate_code, g.gate_type " +
            "from assignments a " +
            "join flights f on f.flight_id = a.flight_id " +
            "join gates g on g.gate_id = a.gate_id " +
            "order by a.assign_id desc";
          try (Connection conn = Condb.getConnection();
               Statement st = conn.createStatement();
               ResultSet rs = st.executeQuery(sql)) {
            while (rs.next()) {
              String status = rs.getString("assign_status");
              String badge = "badge " + ("ACTIVE".equals(status) ? "ok" : "bad");
              String gateLabel = rs.getString("terminal") + " " + rs.getString("gate_code") + " (" + rs.getString("gate_type") + ")";
              String flightLabel = rs.getString("flight_no") + " · " + rs.getString("airline") + " · " + rs.getString("origin") + "→" + rs.getString("destination");
          %>
            <tr>
              <td><a href="assign_edit.jsp?assign_id=<%=rs.getString("assign_id")%>"><%=rs.getString("assign_id")%></a></td>
              <td><%=flightLabel%></td>
              <td><span class="badge purple"><%=gateLabel%></span></td>
              <td><%=rs.getString("occupy_start")%></td>
              <td><%=rs.getString("occupy_end")%></td>
              <td><span class="<%=badge%>"><%=status%></span></td>
              <td><%=rs.getString("note")%></td>
            </tr>
          <%
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

8. assign_validate.js

```
function joinCheck() {
  const frm = document.forms["assign_frm"];
  if (!frm) return true;

  const flightId = (frm.flight_id.value || "").trim();
  const gateId = (frm.gate_id.value || "").trim();
  const occupyStart = (frm.occupy_start.value || "").trim();
  const occupyEnd = (frm.occupy_end.value || "").trim();

  if (!flightId) {
    alert("항공편을 선택해 주세요.");
    frm.flight_id.focus();
    return false;
  }
  if (!gateId) {
    alert("게이트를 선택해 주세요.");
    frm.gate_id.focus();
    return false;
  }
  if (!occupyStart) {
    alert("점유 시작 시간이 입력되지 않았습니다.");
    frm.occupy_start.focus();
    return false;
  }
  if (!occupyEnd) {
    alert("점유 종료 시간이 입력되지 않았습니다.");
    frm.occupy_end.focus();
    return false;
  }
  if (occupyStart >= occupyEnd) {
    alert("점유 시작 시간은 종료 시간보다 이전이어야 합니다.");
    frm.occupy_end.focus();
    return false;
  }

  return true;
}

function search() {
  window.location = "assign_list.jsp";
  return false;
}

function modify() {
  return true;
}
```

9. event_action.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");

String mode = request.getParameter("mode");
String event_id = request.getParameter("event_id");
String flight_id = request.getParameter("flight_id");
String event_type = request.getParameter("event_type");
String event_time = request.getParameter("event_time");
String delay_min = request.getParameter("delay_min");
String reason = request.getParameter("reason");

if (!"insert".equals(mode)) {

response.sendRedirect("event_list.jsp?type=error&msg=mode%20%EB%88%84%E
B%9D%BD");
```

```
  return;
}

String newFlightStatus = null;
if ("DELAYED".equals(event_type)) newFlightStatus = "DELAYED";
else if ("CANCELLED".equals(event_type)) newFlightStatus = "CANCELLED";
else if ("BOARDING".equals(event_type)) newFlightStatus = "BOARDING";
else if ("DEPARTED".equals(event_type)) newFlightStatus = "DEPARTED";
else if ("ARRIVED".equals(event_type)) newFlightStatus = "ARRIVED";

try (Connection conn = Condb.getConnection()) {
  String insertSql = "insert into flight_events (event_id, flight_id,
event_type, event_time, delay_min, reason) values (?,?,?,?,?,?)";
  try (PreparedStatement ps = conn.prepareStatement(insertSql)) {
    ps.setInt(1, Integer.parseInt(event_id));
    ps.setInt(2, Integer.parseInt(flight_id));
    ps.setString(3, event_type);
    ps.setString(4, event_time);
    if (delay_min == null || delay_min.trim().isEmpty()) ps.setNull(5,
Types.NUMERIC);
    else ps.setInt(5, Integer.parseInt(delay_min));
    ps.setString(6, (reason == null || reason.trim().isEmpty()) ? null
: reason);
    ps.executeUpdate();
  }

  if (newFlightStatus != null) {
    try (PreparedStatement ps = conn.prepareStatement("update flights
set status=? where flight_id=?")) {
      ps.setString(1, newFlightStatus);
      ps.setInt(2, Integer.parseInt(flight_id));
      ps.executeUpdate();
    }
  }
} catch (Exception e) {
  e.printStackTrace();

response.sendRedirect("event_list.jsp?type=error&msg=%EC%9D%B4%EB%B2%A4
%ED%8A%B8%20%EB%93%B1%EB%A1%9D%20%EC%8B%A4%ED%8C%A8");
  return;
}
```

```
response.sendRedirect("event_add.jsp?type=success&msg=%EC%9D%B4%EB%B2%A
4%ED%8A%B8%20%EB%93%B1%EB%A1%9D%20%EC%99%84%EB%A3%8C");
%>
```

10. event_add.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");
String event_id = "";
try (Connection conn = Condb.getConnection();
     Statement st = conn.createStatement();
     ResultSet rs = st.executeQuery("select nvl(max(event_id),0)+1
event_id from flight_events")) {
  rs.next();
  event_id = rs.getString("event_id");
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>운항 이벤트 등록</h1>
  <p>지연/탑승/출발/도착/결항 등 이벤트를 기록하고 항공편 상태를 갱신합니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">이벤트 입력</p>
        <p class="card-sub">DELAYED일 때는 지연분과 사유를 함께 입력하는 것을
권장합니다.</p>
      </div>
      <div class="split">
        <a class="btn" href="event_list.jsp">목록</a>
      </div>
    </div>
    <div class="card-body">
      <form method="post" action="event_action.jsp">
```

```
        <input type="hidden" name="mode" value="insert">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">이벤트 ID(자동)</div>
              <input class="input" type="text" name="event_id"
value="<%=event_id%>" readonly>
            </div>
          </div>
          <div class="col-6">
            <div class="field">
              <div class="label">항공편 선택</div>
              <select name="flight_id" required>
                <option value="">선택</option>
                <%
                String fsql = "select flight_id, flight_no, airline,
origin, destination, sched_dep, status from flights order by sched_dep,
flight_id";
                try (Connection conn = Condb.getConnection();
                     Statement st = conn.createStatement();
                     ResultSet rs = st.executeQuery(fsql)) {
                  while (rs.next()) {
                    String fid = rs.getString("flight_id");
                    String label = rs.getString("flight_no") + " · " +
rs.getString("airline")
                        + " · " + rs.getString("origin") + "→" +
rs.getString("destination")
                        + " · " + rs.getString("sched_dep")
                        + " · " + rs.getString("status");
                %>
                <option value="<%=fid%>"><%=label%></option>
                <%
                  }
                } catch (Exception e) {
                  e.printStackTrace();
                }
                %>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">이벤트 유형</div>
              <select name="event_type" required>
                <option value="CHECKIN_OPEN">CHECKIN_OPEN</option>
                <option value="BOARDING">BOARDING</option>
                <option value="DEPARTED">DEPARTED</option>
                <option value="ARRIVED">ARRIVED</option>
                <option value="DELAYED">DELAYED</option>
                <option value="CANCELLED">CANCELLED</option>
                <option value="GATE_CHANGED">GATE_CHANGED</option>
              </select>
            </div>
          </div>

          <div class="col-3">
            <div class="field">
              <div class="label">이벤트 시간</div>
              <input class="input" type="datetime-local"
name="event_time" required>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">지연분(선택)</div>
              <input class="input" type="number" name="delay_min"
min="0" placeholder="예: 30">
            </div>
          </div>
          <div class="col-12">
            <div class="field">
              <div class="label">사유(선택)</div>
              <textarea name="reason" placeholder="예: 기상 악화, 정비
지연, 승객 탑승 지연 등"></textarea>
            </div>
          </div>

          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit">등록</button>
              <button class="btn" type="reset">초기화</button>
            </div>
          </div>
        </div>
      </form>
    </div>
```

```
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

11. event_list.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>이벤트 조회</h1>
  <p>등록된 운항 이벤트를 시간순으로 확인합니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">이벤트 목록</p>
        <p class="card-sub">검색은 즉시 필터링됩니다.</p>
      </div>
      <div class="split">
        <a class="btn primary" href="event_add.jsp">이벤트 등록</a>
      </div>
    </div>
    <div class="card-body">
      <div class="toolbar">
        <div class="split">
          <input class="input w-360" type="text" placeholder="검색:
편명/유형/사유" data-table-filter="eventTable">
        </div>
      </div>

      <div class="table-wrap">
        <table id="eventTable">
          <thead>
            <tr>
              <th>이벤트 ID</th>
              <th>항공편</th>
              <th>유형</th>
              <th>시간</th>
              <th>지연분</th>
              <th>사유</th>
            </tr>
          </thead>
          <tbody>
          <%
          request.setCharacterEncoding("UTF-8");
          String sql =
            "select e.event_id, e.event_type, e.event_time, 
nvl(to_char(e.delay_min),'') delay_min, nvl(e.reason,'') reason, " +
            "f.flight_no, f.airline, f.origin, f.destination " +
            "from flight_events e join flights f on 
f.flight_id=e.flight_id " +
            "order by e.event_time desc, e.event_id desc";
          try (Connection conn = Condb.getConnection();
               Statement st = conn.createStatement();
               ResultSet rs = st.executeQuery(sql)) {
            while (rs.next()) {
              String type = rs.getString("event_type");
              String badge = "badge";
              if ("DELAYED".equals(type)) badge += " warn";
              else if ("CANCELLED".equals(type)) badge += " bad";
              else if ("DEPARTED".equals(type) || 
"ARRIVED".equals(type)) badge += " ok";
              else badge += " purple";
          %>
            <tr>
              <td><%=rs.getString("event_id")%></td>
              <td><%=rs.getString("flight_no")%> · 
<%=rs.getString("airline")%> · 
<%=rs.getString("origin")%>→<%=rs.getString("destination")%></td>
              <td><span class="<%=badge%>"><%=type%></span></td>
              <td><%=rs.getString("event_time")%></td>
              <td><%=rs.getString("delay_min")%></td>
              <td><%=rs.getString("reason")%></td>
            </tr>
          <%
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

12. flight_action.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");

String mode = request.getParameter("mode");
String flight_id = request.getParameter("flight_id");
String flight_no = request.getParameter("flight_no");
String airline = request.getParameter("airline");
String origin = request.getParameter("origin");
String destination = request.getParameter("destination");
String route_type = request.getParameter("route_type");
String sched_dep = request.getParameter("sched_dep");
String sched_arr = request.getParameter("sched_arr");
String status = request.getParameter("status");

if (mode == null) {

response.sendRedirect("flight_list.jsp?type=error&msg=mode%20%EB%88%84%
EB%9D%BD");
  return;
}

try (Connection conn = Condb.getConnection()) {
  if ("insert".equals(mode)) {
    String sql = "insert into flights (flight_id, flight_no, airline,
origin, destination, route_type, sched_dep, sched_arr, status) values
(?,?,?,?,?,?,?,?,?)";
    try (PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setInt(1, Integer.parseInt(flight_id));
      ps.setString(2, flight_no);
      ps.setString(3, airline);
      ps.setString(4, origin);
      ps.setString(5, destination);
      ps.setString(6, route_type);
      ps.setString(7, sched_dep);
      ps.setString(8, (sched_arr == null || sched_arr.trim().isEmpty())
? null : sched_arr);
      ps.setString(9, (status == null || status.trim().isEmpty()) ?
"SCHEDULED" : status);
      ps.executeUpdate();
    }

response.sendRedirect("flight_add.jsp?type=success&msg=%ED%95%AD%EA%B3%
B5%ED%8E%B8%20%EB%93%B1%EB%A1%9D%20%EC%99%84%EB%A3%8C");
    return;
  }

  if ("modify".equals(mode)) {
    String sql = "update flights set flight_no=?, airline=?, origin=?,
destination=?, route_type=?, sched_dep=?, sched_arr=?, status=? where
flight_id=?";
    try (PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setString(1, flight_no);
      ps.setString(2, airline);
      ps.setString(3, origin);
      ps.setString(4, destination);
      ps.setString(5, route_type);
      ps.setString(6, sched_dep);
      ps.setString(7, (sched_arr == null || sched_arr.trim().isEmpty())
? null : sched_arr);
      ps.setString(8, status);
      ps.setInt(9, Integer.parseInt(flight_id));
      ps.executeUpdate();
    }

response.sendRedirect("flight_list.jsp?type=success&msg=%ED%95%AD%EA%B3
%B5%ED%8E%B8%20%EC%88%98%EC%A0%95%20%EC%99%84%EB%A3%8C");
    return;
  }
} catch (Exception e) {
  e.printStackTrace();
```

```
response.sendRedirect("flight_list.jsp?type=error&msg=%EC%B2%98%EB%A6%A
C%20%EC%A4%91%20%EC%98%A4%EB%A5%98%20%EB%B0%9C%EC%83%9D");
  return;
}

response.sendRedirect("flight_list.jsp?type=error&msg=%EC%95%8C%20%EC%8
8%98%20%EC%97%86%EB%8A%94%20mode");
%>
```

13. flight_add.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");
String flight_id = "";
try (Connection conn = Condb.getConnection();
     Statement st = conn.createStatement();
     ResultSet rs = st.executeQuery("select nvl(max(flight_id),0)+1
flight_id from flights")) {
  rs.next();
  flight_id = rs.getString("flight_id");
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>항공편 등록</h1>
  <p>편명/노선/예정시간/상태를 관리합니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">항공편 정보 입력</p>
        <p class="card-sub">예정 출발시간은 배정 시간 자동계산에
사용됩니다.</p>
      </div>
      <div class="split">
        <button class="btn" type="button" onclick="return
search()">목록</button>
      </div>
    </div>
    <div class="card-body">
      <script type="text/javascript" src="flight_validate.js"></script>
      <form method="post" action="flight_action.jsp" name="flight_frm"
onsubmit="return joinCheck()">
        <input type="hidden" name="mode" value="insert">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">항공편 ID(자동)</div>
              <input class="input" type="text" name="flight_id"
value="<%=flight_id%>" readonly>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">편명</div>
              <input class="input" type="text" name="flight_no"
placeholder="예: KE123">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">항공사</div>
              <input class="input" type="text" name="airline"
placeholder="예: KOREAN AIR">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">국내/국제</div>
              <select name="route_type">
                <option value="">선택</option>
                <option value="DOM">DOM(국내)</option>
                <option value="INT">INT(국제)</option>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">출발지</div>
              <input class="input" type="text" name="origin"
placeholder="예: GMP">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">도착지</div>
              <input class="input" type="text" name="destination"
placeholder="예: ICN">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">예정 출발</div>
              <input class="input" type="datetime-local"
name="sched_dep">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">예정 도착(선택)</div>
              <input class="input" type="datetime-local"
name="sched_arr">
            </div>
          </div>

          <div class="col-3">
            <div class="field">
              <div class="label">상태</div>
              <select name="status">
                <option value="SCHEDULED" selected>SCHEDULED</option>
                <option value="BOARDING">BOARDING</option>
                <option value="DEPARTED">DEPARTED</option>
                <option value="ARRIVED">ARRIVED</option>
                <option value="DELAYED">DELAYED</option>
                <option value="CANCELLED">CANCELLED</option>
              </select>
            </div>
          </div>
          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit">등록</button>
              <button class="btn" type="reset">초기화</button>
            </div>
          </div>
        </div>
      </form>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

14. flight_edit.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");

String flight_id = request.getParameter("flight_id");
String flight_no = "";
String airline = "";
String origin = "";
String destination = "";
String route_type = "";
String sched_dep = "";
String sched_arr = "";
String status = "";

try (Connection conn = Condb.getConnection();
     PreparedStatement ps = conn.prepareStatement(
       "select flight_no, airline, origin, destination, route_type,
sched_dep, nvl(sched_arr,'') sched_arr, status from flights where
flight_id=?")) {
  ps.setInt(1, Integer.parseInt(flight_id));
  try (ResultSet rs = ps.executeQuery()) {
    if (rs.next()) {
      flight_no = rs.getString("flight_no");
      airline = rs.getString("airline");
      origin = rs.getString("origin");
      destination = rs.getString("destination");
      route_type = rs.getString("route_type");
      sched_dep = rs.getString("sched_dep");
      sched_arr = rs.getString("sched_arr");
      status = rs.getString("status");
    }
  }
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>항공편 수정</h1>
  <p>편명/시간/상태 변경이 필요할 때 사용합니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">항공편 정보 수정</p>
        <p class="card-sub">항공편 ID는 변경할 수 없습니다.</p>
      </div>
      <div class="split">
        <a class="btn" href="flight_list.jsp">목록</a>
      </div>
    </div>
    <div class="card-body">
      <script type="text/javascript" src="flight_validate.js"></script>
      <form method="post" action="flight_action.jsp" name="flight_frm"
onsubmit="return joinCheck()">
        <input type="hidden" name="mode" value="modify">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">항공편 ID</div>
              <input class="input" type="text" name="flight_id"
value="<%=flight_id%>" readonly>
            </div>
          </div>
          <div class="col-3">
```

```
            <div class="field">
              <div class="label">편명</div>
              <input class="input" type="text" name="flight_no"
value="<%=flight_no%>">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">항공사</div>
              <input class="input" type="text" name="airline"
value="<%=airline%>">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">국내/국제</div>
              <select name="route_type">
                <option value="DOM" <%= "DOM".equals(route_type) ?
"selected" : "" %>>DOM(국내)</option>
                <option value="INT" <%= "INT".equals(route_type) ?
"selected" : "" %>>INT(국제)</option>
              </select>
            </div>
          </div>

          <div class="col-3">
            <div class="field">
              <div class="label">출발지</div>
              <input class="input" type="text" name="origin"
value="<%=origin%>">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">도착지</div>
              <input class="input" type="text" name="destination"
value="<%=destination%>">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">예정 출발</div>
              <input class="input" type="datetime-local"
name="sched_dep" value="<%=sched_dep%>">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">예정 도착(선택)</div>
              <input class="input" type="datetime-local"
name="sched_arr" value="<%=sched_arr%>">
            </div>
          </div>

          <div class="col-3">
            <div class="field">
              <div class="label">상태</div>
              <select name="status">
                <option value="SCHEDULED" <%=
"SCHEDULED".equals(status) ? "selected" : "" %>>SCHEDULED</option>
                <option value="BOARDING" <%= "BOARDING".equals(status)
? "selected" : "" %>>BOARDING</option>
                <option value="DEPARTED" <%= "DEPARTED".equals(status)
? "selected" : "" %>>DEPARTED</option>
                <option value="ARRIVED" <%= "ARRIVED".equals(status) ?
"selected" : "" %>>ARRIVED</option>
                <option value="DELAYED" <%= "DELAYED".equals(status) ?
"selected" : "" %>>DELAYED</option>
                <option value="CANCELLED" <%=
"CANCELLED".equals(status) ? "selected" : "" %>>CANCELLED</option>
              </select>
            </div>
          </div>

          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit" onclick="return
modify()">수정</button>
              <a class="btn" href="flight_list.jsp">취소</a>
            </div>
          </div>
        </div>
      </form>
    </div>
  </div>
```

```
</div>

<jsp:include page="footer.jsp"/>
```

15. flight_list.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>항공편 조회/수정</h1>
  <p>항공편 목록을 확인하고, ID를 클릭하여 수정할 수 있습니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">항공편 목록</p>
        <p class="card-sub">검색은 즉시 필터링됩니다.</p>
      </div>
      <div class="split">
        <a class="btn primary" href="flight_add.jsp">항공편 등록</a>
      </div>
    </div>
    <div class="card-body">
      <div class="toolbar">
        <div class="split">
          <input class="input w-360" type="text" placeholder="검색:
편명/항공사/노선/상태" data-table-filter="flightTable">
        </div>
        <div class="hint">수정하려면 항공편 ID를 클릭</div>
      </div>

      <div class="table-wrap">
        <table id="flightTable">
          <thead>
            <tr>
              <th>항공편 ID</th>
              <th>편명</th>
              <th>항공사</th>
              <th>노선</th>
              <th>구분</th>
              <th>예정 출발</th>
              <th>상태</th>
            </tr>
          </thead>
          <tbody>
          <%
          request.setCharacterEncoding("UTF-8");
          String sql = "select flight_id, flight_no, airline, origin, destination, route_type, sched_dep, status from flights order by sched_dep, flight_id";
          try (Connection conn = Condb.getConnection();
               Statement st = conn.createStatement();
               ResultSet rs = st.executeQuery(sql)) {
            while (rs.next()) {
              String status = rs.getString("status");
              String badge = "badge";
              if ("DELAYED".equals(status)) badge += " warn";
              else if ("CANCELLED".equals(status)) badge += " bad";
              else if ("DEPARTED".equals(status) || "ARRIVED".equals(status)) badge += " ok";
              else badge += " purple";
          %>
            <tr>
              <td><a href="flight_edit.jsp?flight_id=<%=rs.getString("flight_id")%>"><%=rs.getString("flight_id")%></a></td>
              <td><b><%=rs.getString("flight_no")%></b></td>
              <td><%=rs.getString("airline")%></td>
              <td><%=rs.getString("origin")%> → <%=rs.getString("destination")%></td>
              <td><span class="badge"><%=rs.getString("route_type")%></span></td>
              <td><%=rs.getString("sched_dep")%></td>
              <td><span class="<%=badge%>"><%=status%></span></td>
            </tr>
          <%
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

16. flight_validate.js

```
function joinCheck() {
  const frm = document.forms["flight_frm"];
  if (!frm) return true;

  const flightNo = (frm.flight_no.value || "").trim();
  const airline = (frm.airline.value || "").trim();
  const origin = (frm.origin.value || "").trim();
  const destination = (frm.destination.value || "").trim();
  const routeType = (frm.route_type.value || "").trim();
  const schedDep = (frm.sched_dep.value || "").trim();

  if (!flightNo) {
    alert("편명이 입력되지 않았습니다. 예) KE123");
    frm.flight_no.focus();
    return false;
  }
  if (!airline) {
    alert("항공사가 입력되지 않았습니다.");
    frm.airline.focus();
    return false;
  }
  if (!origin) {
    alert("출발지가 입력되지 않았습니다.");
    frm.origin.focus();
    return false;
  }
  if (!destination) {
    alert("도착지가 입력되지 않았습니다.");
    frm.destination.focus();
    return false;
  }
  if (!routeType) {
    alert("국내/국제를 선택해 주세요.");
    frm.route_type.focus();
    return false;
  }
  if (!schedDep) {
    alert("예정 출발시간이 입력되지 않았습니다.");
    frm.sched_dep.focus();
    return false;
  }

  return true;
}

function search() {
  window.location = "flight_list.jsp";
  return false;
}

function modify() {
  return true;
}
```

17. footer.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
  </main>

  <div id="toast" class="toast" role="status" aria-live="polite"
aria-atomic="true">
    <p class="toast-title">알림</p>
    <p class="toast-msg"></p>
  </div>

  <footer class="footer">
    Airport Ops · JSP/Oracle 학습용 프로젝트 · Copyright © DoubleJ
  </footer>
</body>
</html>
```

18. gate_action.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
```

```
<%
request.setCharacterEncoding("UTF-8");

String mode = request.getParameter("mode");
String gate_id = request.getParameter("gate_id");
String gate_code = request.getParameter("gate_code");
String terminal = request.getParameter("terminal");
String gate_type = request.getParameter("gate_type");
String gate_status = request.getParameter("gate_status");

if (mode == null) {

response.sendRedirect("gate_list.jsp?type=error&msg=mode%20%EB%88%84%EB
%9D%BD");
  return;
}

try (Connection conn = Condb.getConnection()) {
  if ("insert".equals(mode)) {
    String sql = "insert into gates (gate_id, gate_code, terminal,
gate_type, gate_status) values (?,?,?,?,?)";
    try (PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setInt(1, Integer.parseInt(gate_id));
      ps.setString(2, gate_code);
      ps.setString(3, terminal);
      ps.setString(4, gate_type);
      ps.setString(5, gate_status);
      ps.executeUpdate();
    }

response.sendRedirect("gate_add.jsp?type=success&msg=%EA%B2%8C%EC%9D%B4
%ED%8A%B8%20%EB%93%B1%EB%A1%9D%20%EC%99%84%EB%A3%8C");
    return;
  }

  if ("modify".equals(mode)) {
    String sql = "update gates set gate_code=?, terminal=?,
gate_type=?, gate_status=? where gate_id=?";
    try (PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setString(1, gate_code);
      ps.setString(2, terminal);
      ps.setString(3, gate_type);
      ps.setString(4, gate_status);
      ps.setInt(5, Integer.parseInt(gate_id));
      ps.executeUpdate();
    }

response.sendRedirect("gate_list.jsp?type=success&msg=%EA%B2%8C%EC%9D%B4%ED%8A%B8%20%EC%88%98%EC%A0%95%20%EC%99%84%EB%A3%8C");
    return;
  }
} catch (Exception e) {
  e.printStackTrace();

response.sendRedirect("gate_list.jsp?type=error&msg=%EC%B2%98%EB%A6%AC%20%EC%A4%91%20%EC%98%A4%EB%A5%98%20%EB%B0%9C%EC%83%9D");
  return;
}

response.sendRedirect("gate_list.jsp?type=error&msg=%EC%95%8C%20%EC%88%98%20%EC%97%86%EB%8A%94%20mode");
%>
```

19. gate_add.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");
String gate_id = "";
try (Connection conn = Condb.getConnection();
     Statement st = conn.createStatement();
     ResultSet rs = st.executeQuery("select nvl(max(gate_id),0)+1 gate_id from gates")) {
  rs.next();
  gate_id = rs.getString("gate_id");
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>게이트 등록</h1>
  <p>터미널/게이트 타입/상태를 관리합니다. 점검(MAINT)·폐쇄(CLOSED) 게이트는 배정이 제한됩니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">게이트 정보 입력</p>
        <p class="card-sub">필수 항목을 입력하고 등록하세요.</p>
      </div>
      <div class="split">
        <button class="btn" type="button" onclick="return search()">목록</button>
      </div>
    </div>
    <div class="card-body">
      <script type="text/javascript" src="gate_validate.js"></script>
      <form method="post" action="gate_action.jsp" name="gate_frm" onsubmit="return joinCheck()">
        <input type="hidden" name="mode" value="insert">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 ID(자동)</div>
              <input class="input" type="text" name="gate_id" value="<%=gate_id%>" readonly>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 코드</div>
              <input class="input" type="text" name="gate_code" placeholder="예: A01">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">터미널</div>
              <select name="terminal">
                <option value="">선택</option>
                <option value="T1">T1</option>
                <option value="T2">T2</option>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 타입</div>
              <select name="gate_type">
                <option value="">선택</option>
                <option value="DOM">DOM(국내)</option>
                <option value="INT">INT(국제)</option>
                <option value="MIX">MIX(혼합)</option>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 상태</div>
              <select name="gate_status">
                <option value="">선택</option>
                <option value="OPEN" selected>OPEN(정상)</option>
                <option value="MAINT">MAINT(점검)</option>
                <option value="CLOSED">CLOSED(폐쇄)</option>
              </select>
            </div>
          </div>
          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit">등록</button>
              <button class="btn" type="reset">초기화</button>
            </div>
          </div>
        </div>
      </form>
      <p class="hint" style="margin-top:10px">
        팁: 코드/터미널/타입을 통일된 규칙으로 입력하면 조회/리포트가 더
깔끔해집니다.
        (표준 정의: <code>docs/gate_codes.md</code>)
      </p>
      <div class="split" style="margin-top:8px">
        <button class="btn" type="button"
data-gate-guide-toggle="#gateGuideAdd">RKSI 표준 게이트 코드 안내</button>
        <a class="btn" href="https://www.ifatc.org/gates?code=RKSI"
target="_blank" rel="noopener">IFATC RKSI 페이지</a>
```

```
      </div>
      <div id="gateGuideAdd" class="guide-panel">
        <h4>RKSI (Incheon) Standard Mapping</h4>
        <ul>
          <li>Concourse Gate 101-132 → <strong>CG101-CG132</strong>
(Airline)</li>
          <li>Maintenance 701-711 → <strong>M701-M711</strong>
(Airline)</li>
          <li>Stand 227-357 → <strong>S227-S357</strong> (Airline)</li>
          <li>Terminal 2 Gate 231-268 → <strong>T2G231-T2G268</strong>
(Airline)</li>
          <li>Terminal Gate 01-50 → <strong>TG01-TG50</strong>
(Airline)</li>
          <li>Cargo 601-647 → <strong>C601-C647</strong> (Cargo)</li>
        </ul>
        <p class="hint">
          더 자세한 지원 기종 목록은 <code>docs/gate_codes.md</code>와
          <a href="https://www.ifatc.org/gates?code=RKSI"
target="_blank" rel="noopener">IFATC RKSI</a>를 참고하세요.
        </p>
      </div>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

20. gate_edit.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");

String gate_id = request.getParameter("gate_id");
String gate_code = "";
String terminal = "";
String gate_type = "";
String gate_status = "";

try (Connection conn = Condb.getConnection();
     PreparedStatement ps = conn.prepareStatement("select gate_code,
terminal, gate_type, gate_status from gates where gate_id=?")) {
  ps.setInt(1, Integer.parseInt(gate_id));
  try (ResultSet rs = ps.executeQuery()) {
    if (rs.next()) {
      gate_code = rs.getString("gate_code");
      terminal = rs.getString("terminal");
      gate_type = rs.getString("gate_type");
      gate_status = rs.getString("gate_status");
    }
  }
} catch (Exception e) {
  e.printStackTrace();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>게이트 수정</h1>
  <p>게이트 상태를 변경하면 배정 가능 여부에 영향을 줍니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">게이트 정보 수정</p>
        <p class="card-sub">게이트 ID는 변경할 수 없습니다.</p>
      </div>
      <div class="split">
        <a class="btn" href="gate_list.jsp">목록</a>
      </div>
    </div>
    <div class="card-body">
      <script type="text/javascript" src="gate_validate.js"></script>
      <form method="post" action="gate_action.jsp" name="gate_frm"
onsubmit="return joinCheck()">
        <input type="hidden" name="mode" value="modify">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 ID</div>
              <input class="input" type="text" name="gate_id"
value="<%=gate_id%>" readonly>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 코드</div>
              <input class="input" type="text" name="gate_code"
value="<%=gate_code%>">
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">터미널</div>
              <select name="terminal">
                <option value="T1" <%= "T1".equals(terminal) ?
"selected" : "" %>>T1</option>
                <option value="T2" <%= "T2".equals(terminal) ?
"selected" : "" %>>T2</option>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 타입</div>
              <select name="gate_type">
                <option value="DOM" <%= "DOM".equals(gate_type) ?
"selected" : "" %>>DOM(국내)</option>
                <option value="INT" <%= "INT".equals(gate_type) ?
"selected" : "" %>>INT(국제)</option>
                <option value="MIX" <%= "MIX".equals(gate_type) ?
"selected" : "" %>>MIX(혼합)</option>
              </select>
            </div>
          </div>
          <div class="col-3">
            <div class="field">
              <div class="label">게이트 상태</div>
              <select name="gate_status">
                <option value="OPEN" <%= "OPEN".equals(gate_status) ?
"selected" : "" %>>OPEN(정상)</option>
                <option value="MAINT" <%= "MAINT".equals(gate_status) ?
"selected" : "" %>>MAINT(점검)</option>
```

```
                <option value="CLOSED" <%= "CLOSED".equals(gate_status) ? "selected" : "" %>>CLOSED(폐쇄)</option>
              </select>
            </div>
          </div>
          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit" onclick="return modify()">수정</button>
              <a class="btn" href="gate_list.jsp">취소</a>
            </div>
          </div>
        </div>
      </form>
      <div class="split" style="margin-top:18px">
        <button class="btn" type="button" data-gate-guide-toggle="#gateGuideEdit">RKSI 표준 게이트 코드 안내</button>
        <a class="btn" href="https://www.ifatc.org/gates?code=RKSI" target="_blank" rel="noopener">IFATC RKSI 페이지</a>
      </div>
      <div id="gateGuideEdit" class="guide-panel">
        <h4>RKSI Gate Naming Quick View</h4>
        <ul>
          <li>Concourse Gate 101-132 → <strong>CG101-CG132</strong></li>
          <li>Maintenance 701-711 → <strong>M701-M711</strong></li>
          <li>Stand 227-357 → <strong>S227-S357</strong></li>
          <li>Terminal 2 Gate 231-268 → <strong>T2G231-T2G268</strong></li>
          <li>Terminal Gate 01-50 → <strong>TG01-TG50</strong></li>
          <li>Cargo 601-647 → <strong>C601-C647</strong></li>
        </ul>
        <p class="hint">
          전체 매핑 및 지원 기준: <code>docs/gate_codes.md</code>,
          <a href="https://www.ifatc.org/gates?code=RKSI" target="_blank" rel="noopener">IFATC RKSI</a>
        </p>
      </div>
    </div>
  </div>
</div>
```

```
<jsp:include page="footer.jsp"/>
```

21. gate_list.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>게이트 조회/수정</h1>
  <p>게이트 목록을 확인하고, 클릭하여 수정할 수 있습니다.</p>
</div>

<div class="grid">
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">게이트 목록</p>
        <p class="card-sub">검색은 클라이언트 필터(즉시 반영)로 동작합니다.</p>
      </div>
      <div class="split">
        <a class="btn primary" href="gate_add.jsp">게이트 등록</a>
      </div>
    </div>
    <div class="card-body">
      <div class="toolbar">
        <div class="split">
          <input class="input w-360" type="text" placeholder="검색: 코드/터미널/타입/상태" data-table-filter="gateTable">
        </div>
        <div class="hint">수정하려면 게이트 ID를 클릭</div>
      </div>

      <div class="table-wrap">
        <table id="gateTable">
          <thead>
            <tr>
              <th>게이트 ID</th>
              <th>코드</th>
              <th>터미널</th>
              <th>타입</th>
              <th>상태</th>
            </tr>
          </thead>
          <tbody>
          <%
          request.setCharacterEncoding("UTF-8");
          try (Connection conn = Condb.getConnection();
               Statement st = conn.createStatement();
               ResultSet rs = st.executeQuery("select gate_id,
gate_code, terminal, gate_type, gate_status from gates order by
gate_id")) {
            while (rs.next()) {
              String status = rs.getString("gate_status");
              String badge = "badge";
              if ("OPEN".equals(status)) badge += " ok";
              else if ("MAINT".equals(status)) badge += " warn";
              else badge += " bad";
          %>
            <tr>
              <td><a
href="gate_edit.jsp?gate_id=<%=rs.getString("gate_id")%>"><%=rs.getStri
ng("gate_id")%></a></td>
              <td><%=rs.getString("gate_code")%></td>
              <td><span class="badge
purple"><%=rs.getString("terminal")%></span></td>
              <td><span
class="badge"><%=rs.getString("gate_type")%></span></td>
              <td><span class="<%=badge%>"><%=status%></span></td>
            </tr>
          <%
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

22. gate_validate.js

```
function joinCheck() {
  const frm = document.forms["gate_frm"];
  if (!frm) return true;

  const gateCode = (frm.gate_code.value || "").trim();
  const terminal = (frm.terminal.value || "").trim();
  const gateType = (frm.gate_type.value || "").trim();
  const gateStatus = (frm.gate_status.value || "").trim();

  if (!gateCode) {
    alert("게이트 코드가 입력되지 않았습니다. 예) A01");
    frm.gate_code.focus();
    return false;
  }
  if (!terminal) {
    alert("터미널을 선택해 주세요.");
    frm.terminal.focus();
    return false;
  }
  if (!gateType) {
    alert("게이트 타입을 선택해 주세요.");
    frm.gate_type.focus();
    return false;
  }
  if (!gateStatus) {
    alert("게이트 상태를 선택해 주세요.");
    frm.gate_status.focus();
    return false;
  }
  return true;
}

function search() {
  window.location = "gate_list.jsp";
  return false;
}

function modify() {
  return true;
}
```

23. header.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!doctype html>
<html lang="ko" data-theme="dark">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>공항 항공편/게이트 운영 관리</title>
  <link rel="stylesheet" href="app.css">
  <script defer src="app.js"></script>
</head>
<body>
  <div class="topbar">
    <div class="topbar-inner">
      <a class="brand" href="index.jsp">
        <span class="logo" aria-hidden="true"></span>
        <span>Airport Ops</span>
      </a>

      <nav class="nav" aria-label="main">
        <a class="chip" data-page="gate_add.jsp"
href="gate_add.jsp">게이트 등록</a>
        <a class="chip" data-page="gate_list.jsp"
href="gate_list.jsp">게이트 조회</a>
        <a class="chip" data-page="flight_add.jsp"
href="flight_add.jsp">항공편 등록</a>
        <a class="chip" data-page="flight_list.jsp"
href="flight_list.jsp">항공편 조회</a>
        <a class="chip" data-page="assign_add.jsp"
href="assign_add.jsp">게이트 배정</a>
        <a class="chip" data-page="assign_list.jsp"
href="assign_list.jsp">배정 조회</a>
        <a class="chip" data-page="event_add.jsp"
href="event_add.jsp">이벤트 등록</a>
        <a class="chip" data-page="event_list.jsp"
href="event_list.jsp">이벤트 조회</a>
        <a class="chip" data-page="reports.jsp"
href="reports.jsp">리포트</a>
      </nav>

      <div class="actions">
```

```
        <button class="btn" type="button"
data-theme-toggle>테마</button>
        <a class="btn primary" href="index.jsp">홈</a>
      </div>
    </div>
  </div>

  <main class="container">
```

24. index.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>공항 항공편/게이트 운영 관리</h1>
  <p>
    항공편(Flight) · 게이트(Gate) · 배정(Assignment) · 운항 이벤트(Event)를
관리하고,
    일일 운항표/지연 통계/게이트 사용률을 리포트로 조회하는 프로그램입니다.
  </p>
</div>

<div class="grid">
  <div class="card" style="grid-column: span 12;">
    <div class="card-header">
      <div>
        <p class="card-title">시작하기</p>
        <p class="card-sub">권장 흐름: 게이트 → 항공편 → 배정 → 이벤트 →
리포트</p>
      </div>
      <div class="split">
        <a class="btn primary" href="gate_add.jsp">게이트 등록</a>
        <a class="btn primary" href="flight_add.jsp">항공편 등록</a>
        <a class="btn primary" href="assign_add.jsp">게이트 배정</a>
        <a class="btn" href="reports.jsp">리포트</a>
      </div>
    </div>
    <div class="card-body">
      <div class="row">
        <div class="col-6">
          <p class="hint">
```

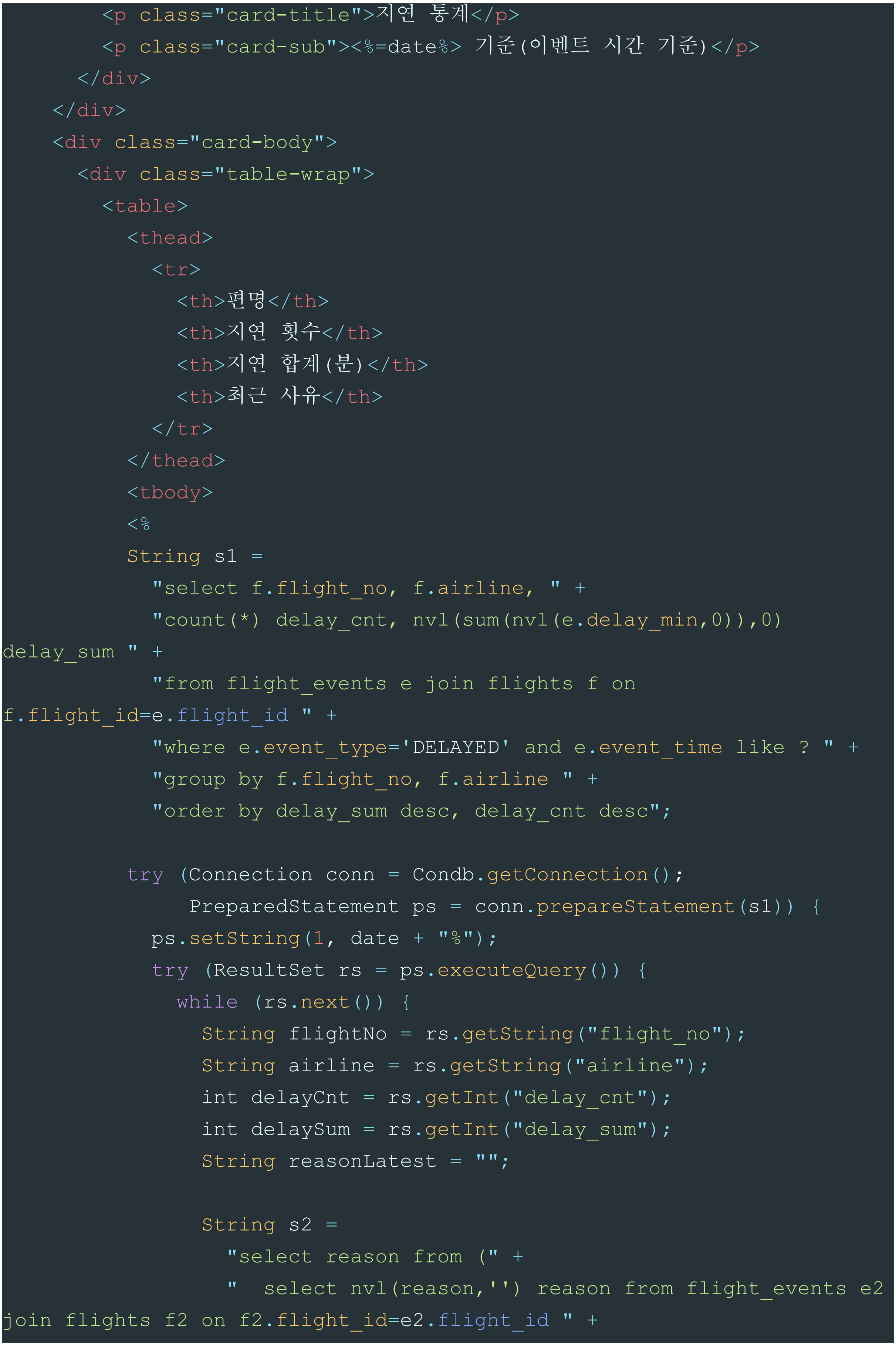
```
            DB 스키마는 <b><code>db/airport_schema.sql</code></b> 을
사용합니다.
            (기존 서점 테이블과 별개로 생성)
          </p>
        </div>
        <div class="col-6">
          <p class="hint">
            게이트 배정은 <b>시간 구간 중복</b>이 있으면 등록/변경이 차단됩니다.
            (출발 60분 전 ~ 출발 15분 후 자동 계산 지원)
          </p>
        </div>
      </div>
    </div>
  </div>
</div>

<jsp:include page="footer.jsp"/>
```

25. reports.jsp

```
<%@page import="airport.Condb"%>
<%@page import="java.sql.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%
request.setCharacterEncoding("UTF-8");
String view = request.getParameter("view");
if (view == null || view.trim().isEmpty()) view = "daily";

String date = request.getParameter("date");
if (date == null || date.trim().isEmpty()) {
  // 'YYYY-MM-DD' (서버 로컬 기준)
  java.time.LocalDate today = java.time.LocalDate.now();
  date = today.toString();
}
%>
<jsp:include page="header.jsp"/>

<div class="hero">
  <h1>리포트</h1>
  <p>일일 운항표 · 지연 통계 · 게이트 사용률을 확인합니다.</p>
</div>

<div class="grid">
```

```
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">리포트 선택</p>
        <p class="card-sub">날짜 기준으로 조회합니다.</p>
      </div>
      <div class="split">
        <a class="btn <%= "daily".equals(view) ? "primary" : "" %>" href="reports.jsp?view=daily&date=<%=date%>">일일 운항표</a>
        <a class="btn <%= "delay".equals(view) ? "primary" : "" %>" href="reports.jsp?view=delay&date=<%=date%>">지연 통계</a>
        <a class="btn <%= "gate".equals(view) ? "primary" : "" %>" href="reports.jsp?view=gate&date=<%=date%>">게이트 사용률</a>
      </div>
    </div>
    <div class="card-body">
      <form method="get" action="reports.jsp">
        <input type="hidden" name="view" value="<%=view%>">
        <div class="row">
          <div class="col-3">
            <div class="field">
              <div class="label">조회 날짜</div>
              <input class="input" type="date" name="date" value="<%=date%>">
            </div>
          </div>
          <div class="col-12">
            <div class="split">
              <button class="btn primary" type="submit">조회</button>
              <span class="hint">항공편의 `sched_dep`가 <%=date%>로 시작하는 데이터 기준</span>
            </div>
          </div>
        </div>
      </form>
    </div>
  </div>

  <%
  if ("daily".equals(view)) {
  %>
  <div class="card">
    <div class="card-header">
```

```
      <div>
        <p class="card-title">일일 운항표</p>
        <p class="card-sub"><%=date%> 기준</p>
      </div>
      <div class="hint">게이트는 ACTIVE 배정 중 항공편별 최신 배정 1건을
표시</div>
    </div>
    <div class="card-body">
      <div class="table-wrap">
        <table>
          <thead>
            <tr>
              <th>편명</th>
              <th>노선</th>
              <th>예정 출발</th>
              <th>구분</th>
              <th>게이트</th>
              <th>상태</th>
            </tr>
          </thead>
          <tbody>
          <%
          String dsql =
            "select f.flight_no, f.airline, f.origin, f.destination,
f.sched_dep, f.route_type, f.status, " +
            "nvl(g.terminal,'') terminal, nvl(g.gate_code,'')
gate_code, nvl(g.gate_type,'') gate_type " +
            "from flights f " +
            "left join (" +
            "  select a.flight_id, max(a.assign_id) max_assign_id from
assignments a " +
            "  where a.assign_status='ACTIVE' group by a.flight_id" +
            ") am on am.flight_id = f.flight_id " +
            "left join assignments a2 on a2.assign_id =
am.max_assign_id " +
            "left join gates g on g.gate_id = a2.gate_id " +
            "where f.sched_dep like ? " +
            "order by f.sched_dep, f.flight_id";

          try (Connection conn = Condb.getConnection();
               PreparedStatement ps = conn.prepareStatement(dsql)) {
            ps.setString(1, date + "%");
            try (ResultSet rs = ps.executeQuery()) {
              while (rs.next()) {
                String status = rs.getString("status");
                String badge = "badge";
                if ("DELAYED".equals(status)) badge += " warn";
                else if ("CANCELLED".equals(status)) badge += " bad";
                else if ("DEPARTED".equals(status) ||
"ARRIVED".equals(status)) badge += " ok";
                else badge += " purple";
                String terminal = rs.getString("terminal");
                String gateLabel = (terminal == null ||
terminal.trim().isEmpty()) ? "-" :
                  (terminal + " " + rs.getString("gate_code") + " (" +
rs.getString("gate_type") + ")");
          %>
            <tr>
              <td><b><%=rs.getString("flight_no")%></b> ·
<%=rs.getString("airline")%></td>
              <td><%=rs.getString("origin")%> →
<%=rs.getString("destination")%></td>
              <td><%=rs.getString("sched_dep")%></td>
              <td><span
class="badge"><%=rs.getString("route_type")%></span></td>
              <td><span class="badge purple"><%=gateLabel%></span></td>
              <td><span class="<%=badge%>"><%=status%></span></td>
            </tr>
          <%
              }
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
  <%
  } else if ("delay".equals(view)) {
  %>
  <div class="card">
    <div class="card-header">
      <div>
```

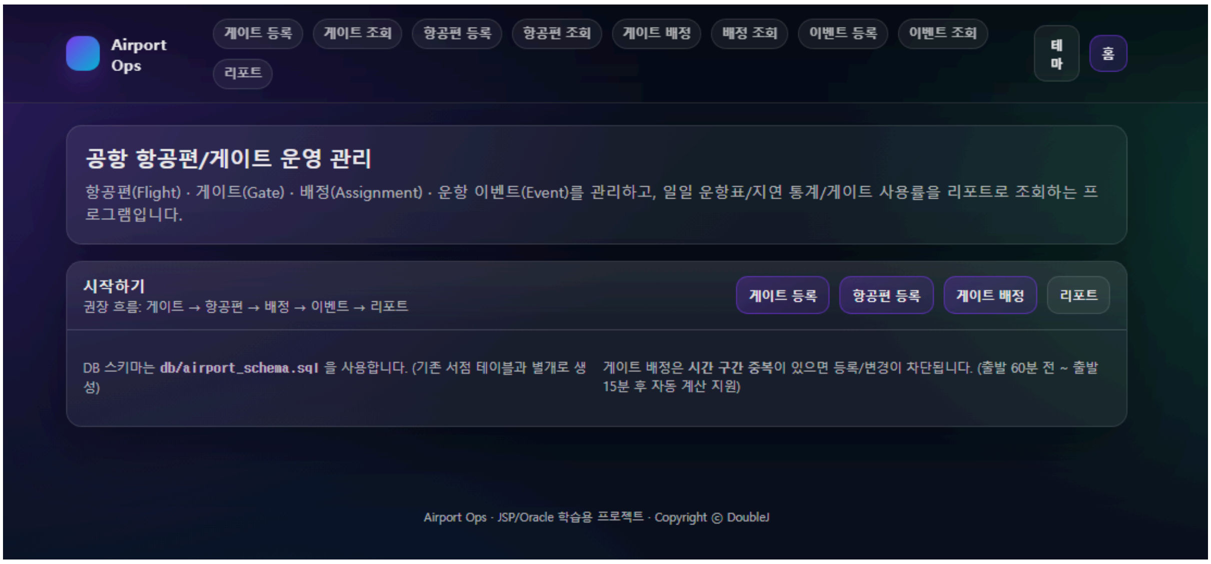


```
        <p class="card-title">지연 통계</p>
        <p class="card-sub"><%=date%> 기준(이벤트 시간 기준)</p>
      </div>
    </div>
    <div class="card-body">
      <div class="table-wrap">
        <table>
          <thead>
            <tr>
              <th>편명</th>
              <th>지연 횟수</th>
              <th>지연 합계(분)</th>
              <th>최근 사유</th>
            </tr>
          </thead>
          <tbody>
          <%
          String s1 =
            "select f.flight_no, f.airline, " +
            "count(*) delay_cnt, nvl(sum(nvl(e.delay_min,0)),0)
delay_sum " +
            "from flight_events e join flights f on
f.flight_id=e.flight_id " +
            "where e.event_type='DELAYED' and e.event_time like ? " +
            "group by f.flight_no, f.airline " +
            "order by delay_sum desc, delay_cnt desc";

          try (Connection conn = Condb.getConnection();
               PreparedStatement ps = conn.prepareStatement(s1)) {
            ps.setString(1, date + "%");
            try (ResultSet rs = ps.executeQuery()) {
              while (rs.next()) {
                String flightNo = rs.getString("flight_no");
                String airline = rs.getString("airline");
                int delayCnt = rs.getInt("delay_cnt");
                int delaySum = rs.getInt("delay_sum");
                String reasonLatest = "";

                String s2 =
                  "select reason from (" +
                  "  select nvl(reason,'') reason from flight_events e2
join flights f2 on f2.flight_id=e2.flight_id " +
                  "  where e2.event_type='DELAYED' and f2.flight_no=? and e2.event_time like ? " +
                  "  order by e2.event_time desc, e2.event_id desc" +
                  ") where rownum=1";
                try (PreparedStatement ps2 = conn.prepareStatement(s2)) {
                  ps2.setString(1, flightNo);
                  ps2.setString(2, date + "%");
                  try (ResultSet rs2 = ps2.executeQuery()) {
                    if (rs2.next()) reasonLatest = rs2.getString(1);
                  }
                }
          %>
            <tr>
              <td><b><%=flightNo%></b> · <%=airline%></td>
              <td><span class="badge warn"><%=delayCnt%></span></td>
              <td><span class="badge warn"><%=delaySum%></span></td>
              <td><%=reasonLatest%></td>
            </tr>
          <%
              }
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
  <%
  } else if ("gate".equals(view)) {
  %>
  <div class="card">
    <div class="card-header">
      <div>
        <p class="card-title">게이트 사용률</p>
        <p class="card-sub"><%=date%> 기준(점유 시작이 해당 날짜인 ACTIVE 배정)</p>
      </div>
    </div>
    <div class="card-body">
```

```
      <div class="table-wrap">
        <table>
          <thead>
            <tr>
              <th>게이트</th>
              <th>배정 건수</th>
              <th>상태</th>
            </tr>
          </thead>
          <tbody>
          <%
          String gsql =
            "select g.terminal, g.gate_code, g.gate_type, g.gate_status, count(*) cnt " +
            "from assignments a join gates g on g.gate_id=a.gate_id " +
            "where a.assign_status='ACTIVE' and a.occupy_start like ? " +
            "group by g.terminal, g.gate_code, g.gate_type, g.gate_status " +
            "order by cnt desc, g.terminal, g.gate_code";
          try (Connection conn = Condb.getConnection();
               PreparedStatement ps = conn.prepareStatement(gsql)) {
            ps.setString(1, date + "%");
            try (ResultSet rs = ps.executeQuery()) {
              while (rs.next()) {
                String status = rs.getString("gate_status");
                String badge = "badge";
                if ("OPEN".equals(status)) badge += " ok";
                else if ("MAINT".equals(status)) badge += " warn";
                else badge += " bad";
          %>
            <tr>
              <td><%=rs.getString("terminal")%> <b><%=rs.getString("gate_code")%></b> (<%=rs.getString("gate_type")%>)</td>
              <td><span class="badge purple"><%=rs.getInt("cnt")%></span></td>
              <td><span class="<%=badge%>"><%=status%></span></td>
            </tr>
          <%
              }
            }
          } catch (Exception e) {
            e.printStackTrace();
          }
          %>
          </tbody>
        </table>
      </div>
    </div>
  </div>
  <%
  }
  %>
</div>

<jsp:include page="footer.jsp"/>
```

26. java/airport/Condb.java

```
package airport;

import java.sql.Connection;
import java.sql.DriverManager;

public class Condb {
    public static Connection getConnection() throws Exception {
        Class.forName("oracle.jdbc.OracleDriver");
        String url = "jdbc:oracle:thin:@//localhost:1521/xe";
        String user = "system";
        String password = "1234";
        return DriverManager.getConnection(url, user, password);
    }
}
```



공항 항공편/게이트 운영 관리

항공편(Flight) · 게이트(Gate) · 배정(Assignment) · 운항 이벤트(Event)를 관리하고, 일일 운항표/지연 통계/게이트 사용률을 리포트로 조회하는 프로그램입니다.

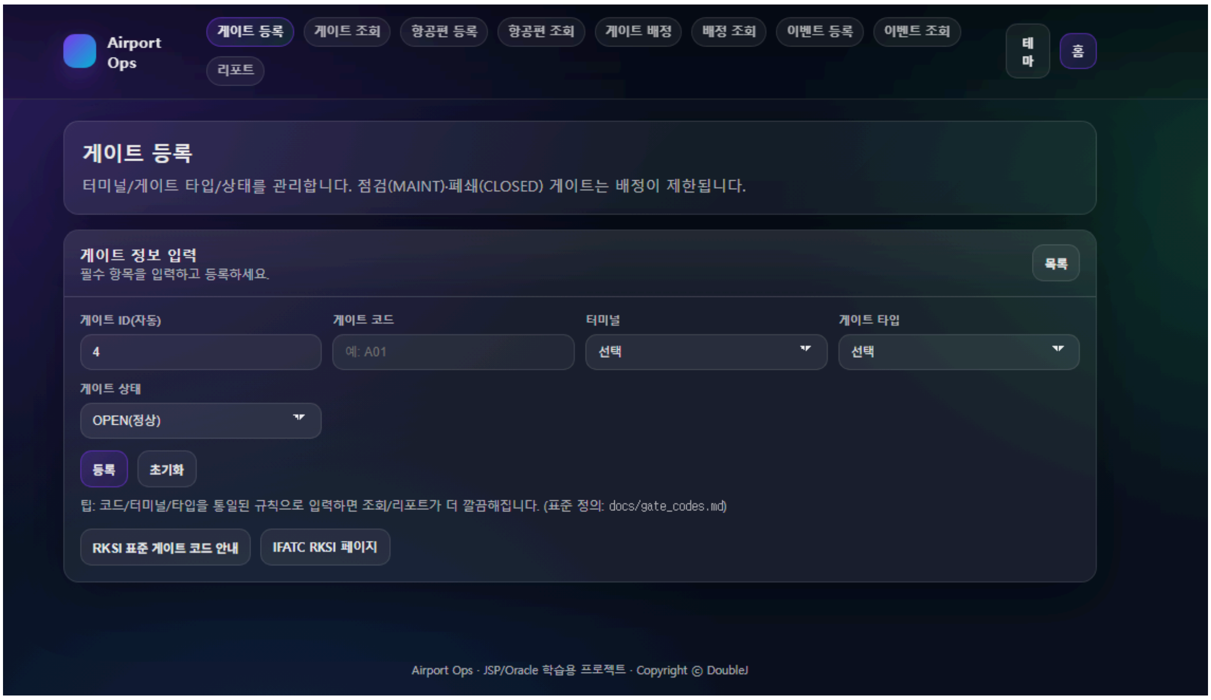
시작하기

권장 흐름: 게이트 → 항공편 → 배정 → 이벤트 → 리포트

DB 스키마는 **db/airport_schema.sql** 을 사용합니다. (기존 서점 테이블과 별개로 생성)

게이트 배정은 **시간 구간 중복**이 있으면 등록/변경이 차단됩니다. (출발 60분 전 ~ 출발 15분 후 자동 계산 지원)

Airport Ops · JSP/Oracle 학습용 프로젝트 · Copyright © DoubleJ



게이트 등록

터미널/게이트 타입/상태를 관리합니다. 점검(MAINT)·폐쇄(CLOSED) 게이트는 배정이 제한됩니다.

게이트 정보 입력

필수 항목을 입력하고 등록하세요.

게이트 ID(자동): 4

게이트 코드: 예: A01

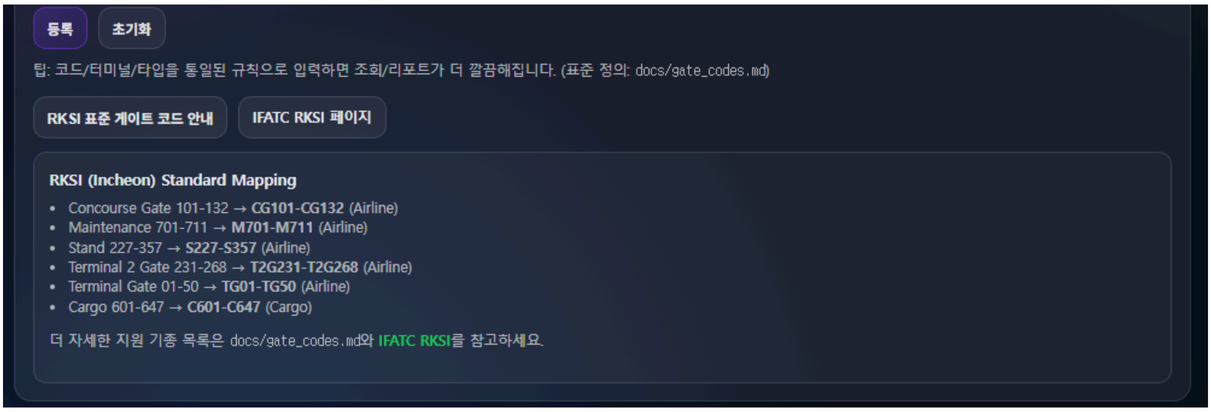
터미널: 선택

게이트 타입: 선택

게이트 상태: OPEN(정상)

팁: 코드/터미널/타입을 통일된 규칙으로 입력하면 조회/리포트가 더 깔끔해집니다. (표준 정의: docs/gate_codes.md)

Airport Ops · JSP/Oracle 학습용 프로젝트 · Copyright © DoubleJ

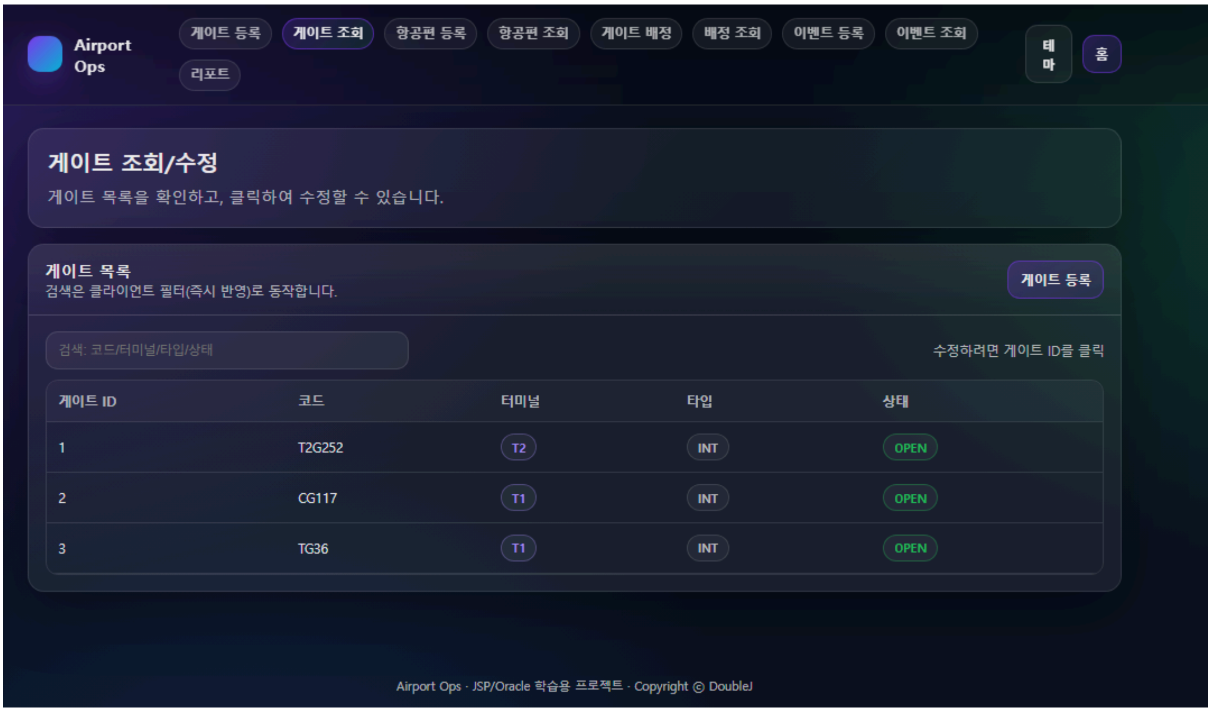


팁: 코드/터미널/타입을 통일된 규칙으로 입력하면 조회/리포트가 더 깔끔해집니다. (표준 정의: docs/gate_codes.md)

RKSI (Incheon) Standard Mapping

- Concourse Gate 101-132 → **CG101-CG132** (Airline)
- Maintenance 701-711 → **M701-M711** (Airline)
- Stand 227-357 → **S227-S357** (Airline)
- Terminal 2 Gate 231-268 → **T2G231-T2G268** (Airline)
- Terminal Gate 01-50 → **TG01-TG50** (Airline)
- Cargo 601-647 → **C601-C647** (Cargo)

더 자세한 지원 기종 목록은 docs/gate_codes.md와 IFATC RKSI를 참고하세요.



게이트 조회/수정

게이트 목록을 확인하고, 클릭하여 수정할 수 있습니다.

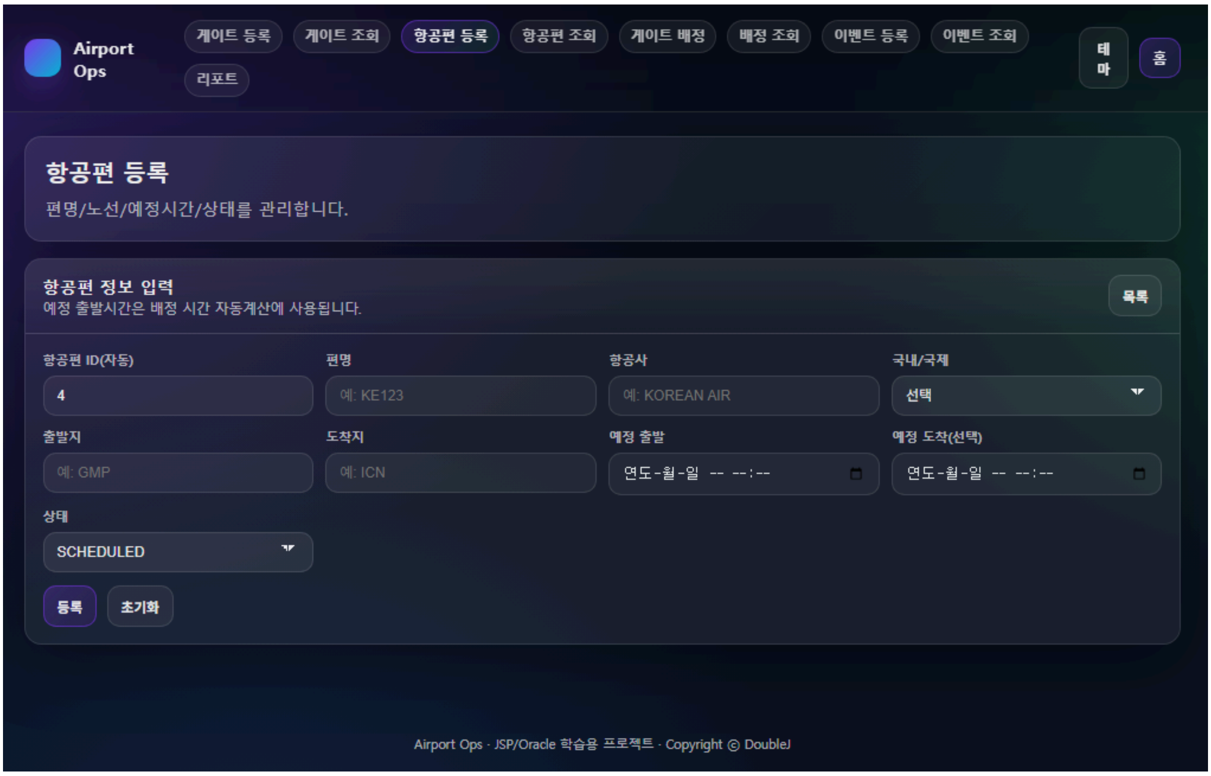
게이트 목록

검색은 클라이언트 필터(즉시 반영)로 동작합니다.

수정하려면 게이트 ID를 클릭

게이트 ID	코드	터미널	타입	상태
1	T2G252	T2	INT	OPEN
2	CG117	T1	INT	OPEN
3	TG36	T1	INT	OPEN

Airport Ops · JSP/Oracle 학습용 프로젝트 · Copyright © DoubleJ



항공편 등록

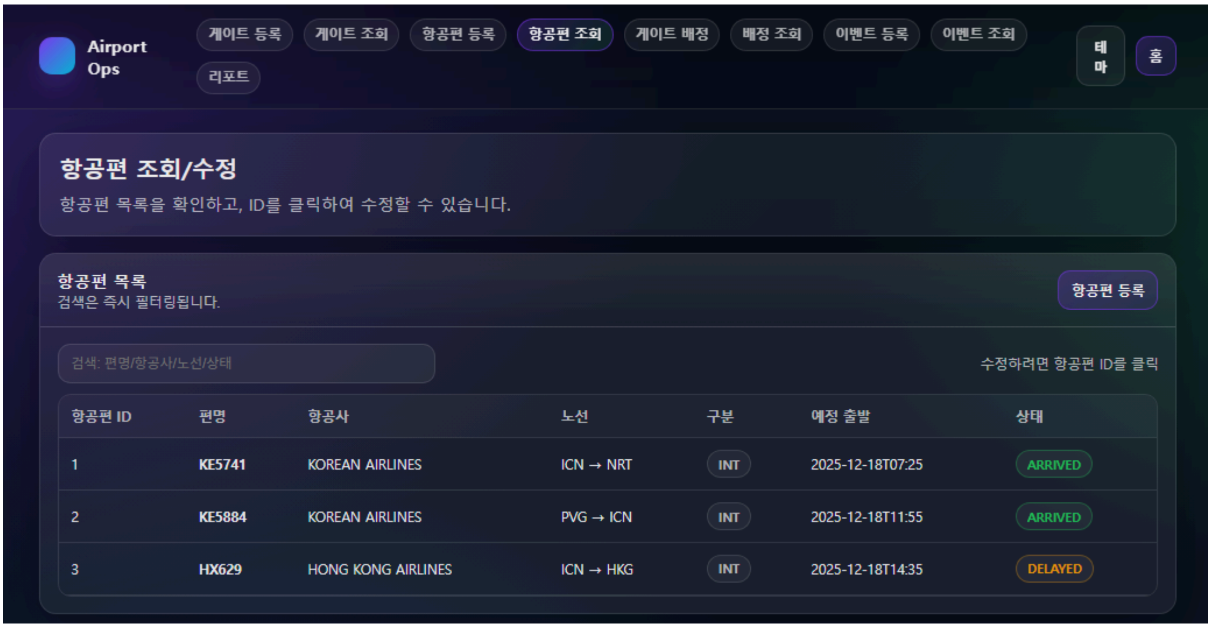
편명/노선/예정시간/상태를 관리합니다.

항공편 정보 입력

예정 출발시간은 배정 시간 자동계산에 사용됩니다.

항공편 ID(자동)	편명	항공사	국내/국제
4	예: KE123	예: KOREAN AIR	선택
출발지	도착지	예정 출발	예정 도착(선택)
예: GMP	예: ICN	연도-월-일 -- --:--	연도-월-일 -- --:--
상태			
SCHEDULED			

Airport Ops · JSP/Oracle 학습용 프로젝트 · Copyright © DoubleJ



항공편 조회/수정

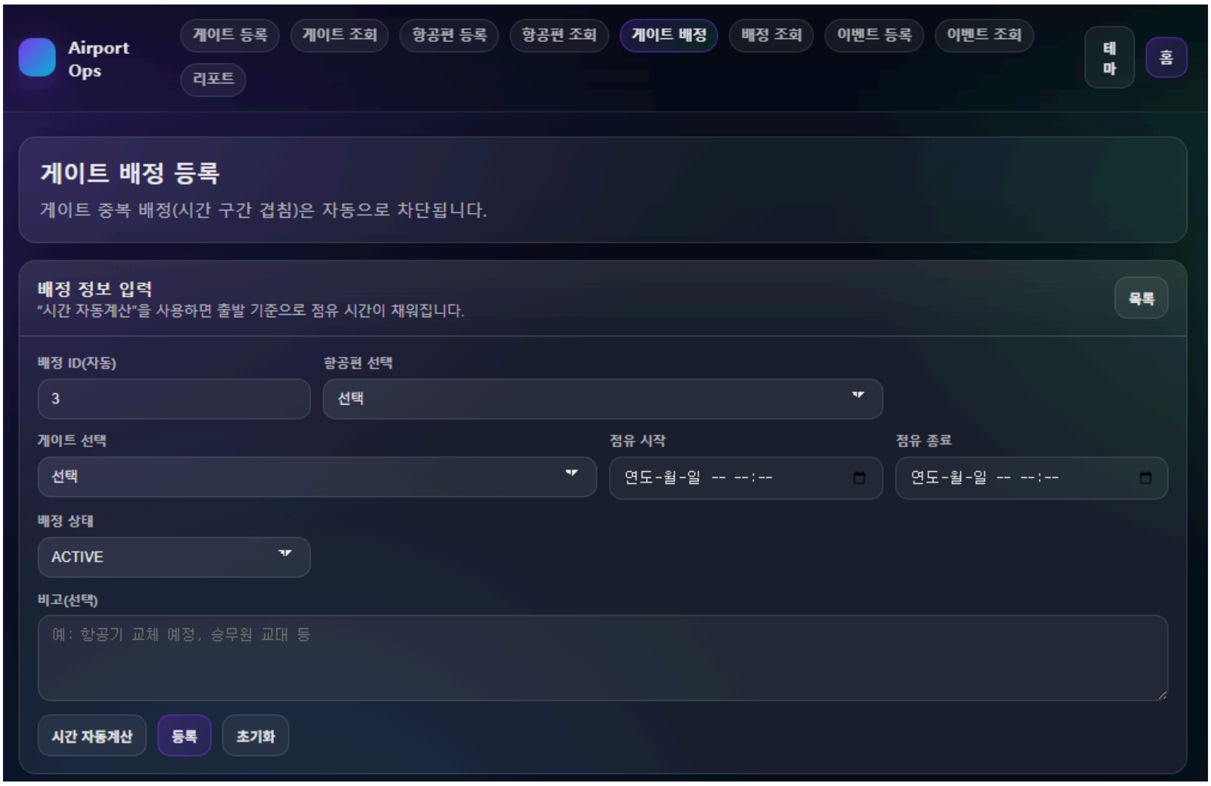
항공편 목록을 확인하고, ID를 클릭하여 수정할 수 있습니다.

항공편 목록

검색은 즉시 필터링됩니다.

수정하려면 항공편 ID를 클릭

항공편 ID	편명	항공사	노선	구분	예정 출발	상태
1	KE5741	KOREAN AIRLINES	ICN → NRT	INT	2025-12-18T07:25	ARRIVED
2	KE5884	KOREAN AIRLINES	PVG → ICN	INT	2025-12-18T11:55	ARRIVED
3	HX629	HONG KONG AIRLINES	ICN → HKG	INT	2025-12-18T14:35	DELAYED



게이트 배정 등록

게이트 중복 배정(시간 구간 겹침)은 자동으로 차단됩니다.

배정 정보 입력

"시간 자동계산"을 사용하면 출발 기준으로 점유 시간이 채워집니다.

배정 ID(자동): 3

항공편 선택: 선택

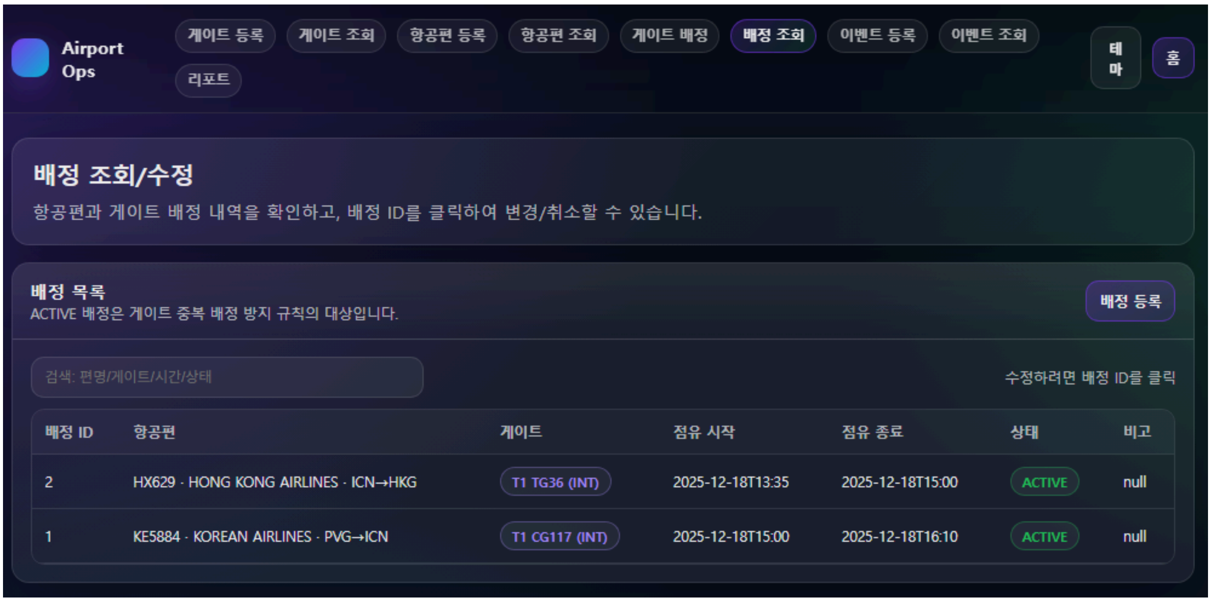
게이트 선택: 선택

점유 시작: 연도-월-일 -- --:--

점유 종료: 연도-월-일 -- --:--

배정 상태: ACTIVE

비고(선택): 예: 항공기 교체 예정, 승무원 교대 등



배정 조회/수정

항공편과 게이트 배정 내역을 확인하고, 배정 ID를 클릭하여 변경/취소할 수 있습니다.

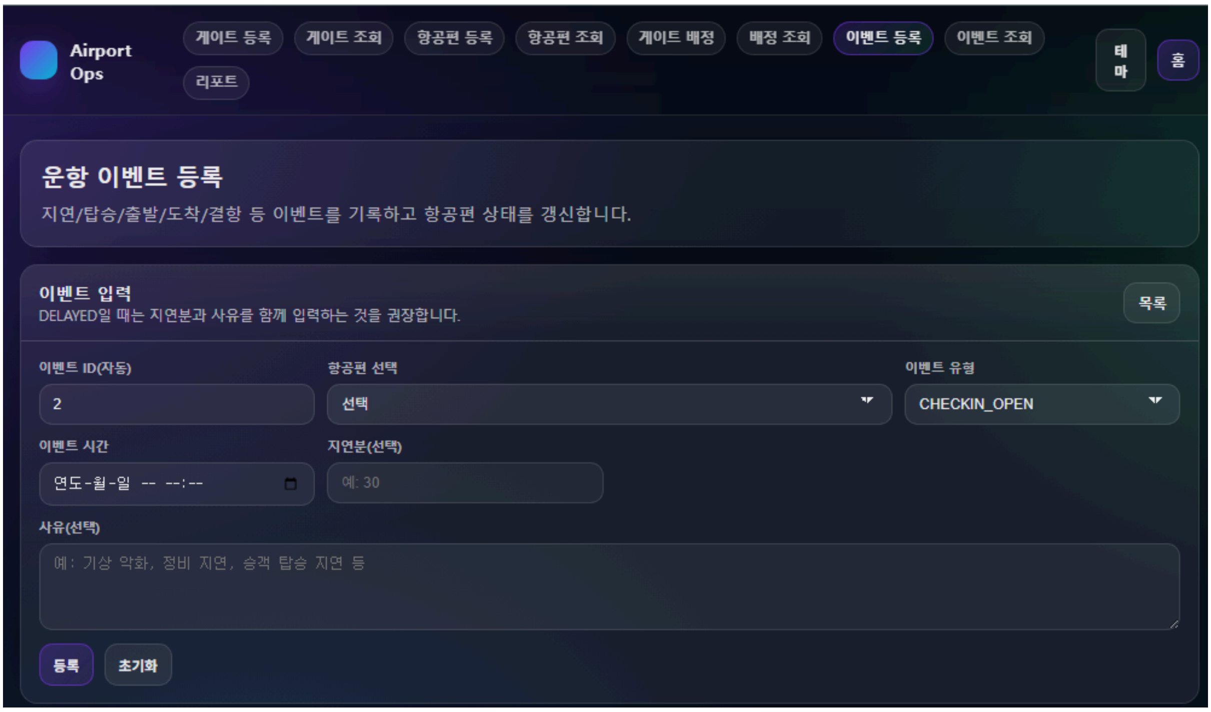
배정 목록

ACTIVE 배정은 게이트 중복 배정 방지 규칙의 대상입니다.

검색: 편명/게이트/시간/상태

수정하려면 배정 ID를 클릭

배정 ID	항공편	게이트	점유 시작	점유 종료	상태	비고
2	HX629 · HONG KONG AIRLINES · ICN→HKG	T1 TG36 (INT)	2025-12-18T13:35	2025-12-18T15:00	ACTIVE	null
1	KE5884 · KOREAN AIRLINES · PVG→ICN	T1 CG117 (INT)	2025-12-18T15:00	2025-12-18T16:10	ACTIVE	null



운항 이벤트 등록

지연/탑승/출발/도착/결항 등 이벤트를 기록하고 항공편 상태를 갱신합니다.

이벤트 입력

DELAYED일 때는 지연분과 사유를 함께 입력하는 것을 권장합니다.

이벤트 ID(자동): 2

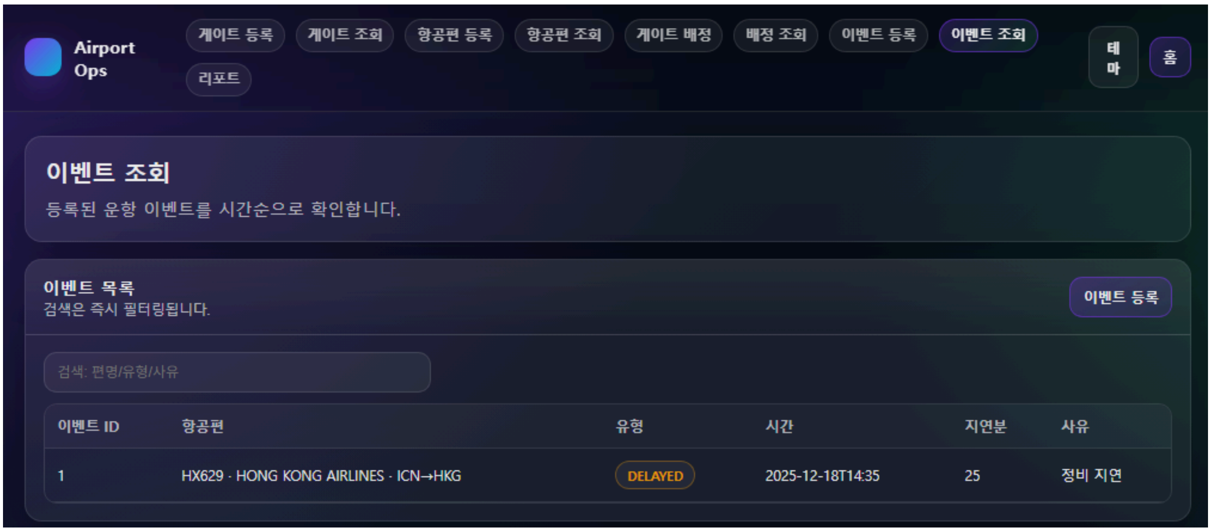
항공편 선택: 선택

이벤트 유형: CHECKIN_OPEN

이벤트 시간: 연도-월-일 -- --:--

지연분(선택): 예: 30

사유(선택): 예: 기상 악화, 정비 지연, 승객 탑승 지연 등



이벤트 조회

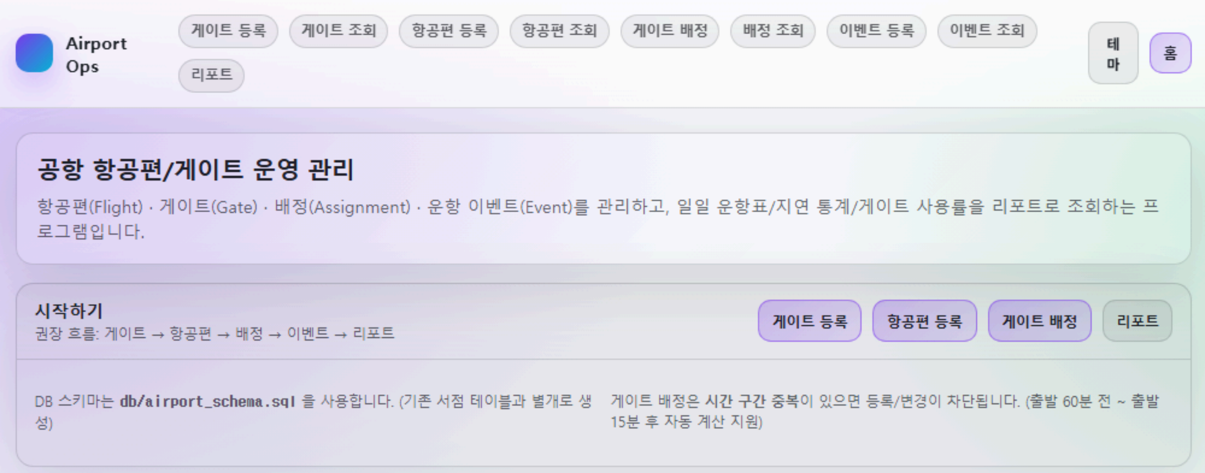
등록된 운항 이벤트를 시간순으로 확인합니다.

이벤트 목록

검색은 즉시 필터링됩니다.

검색: 편명/유형/사유

이벤트 ID	항공편	유형	시간	지연분	사유
1	HX629 · HONG KONG AIRLINES · ICN→HKG	DELAYED	2025-12-18T14:35	25	정비 지연



공항 항공편/게이트 운영 관리

항공편(Flight) · 게이트(Gate) · 배정(Assignment) · 운항 이벤트(Event)를 관리하고, 일일 운항표/지연 통계/게이트 사용률을 리포트로 조회하는 프로그램입니다.

시작하기

권장 흐름: 게이트 → 항공편 → 배정 → 이벤트 → 리포트

DB 스키마는 `db/airport_schema.sql` 을 사용합니다. (기존 서점 테이블과 별개로 생성)

게이트 배정은 **시간 구간 중복**이 있으면 등록/변경이 차단됩니다. (출발 60분 전 ~ 출발 15분 후 자동 계산 지원)