

**a) Seleção de Ingressos com Detalhes de Evento e Torcedor, Ordenados por Preço**  
**Junções de 3 ou mais tabelas, com ORDER BY:**

```
SELECT
    Ingresso.id_ingresso,
    Ingresso.num_assento,
    Ingresso.preco,
    Ingresso.setor,
    Evento.local AS local_evento,
    Evento.data AS data_evento,
    Torcedor.nome AS nome_torcedor,
    Torcedor.pais AS pais_torcedor
FROM
    Ingresso
JOIN
    Evento ON Ingresso.id_evento = Evento.id_evento
JOIN
    Torcedor ON Ingresso.num_passaporte_torcedor = Torcedor.num_passaporte_torcedor
ORDER BY
    Ingresso.preco DESC;
```

**b) Seleção de Ingressos com Detalhes de Evento e Torcedor, Filtrados por Preço e Ordenados por Preço**

**Junções de 3 ou mais tabelas, com ORDER BY e filtros na cláusula WHERE:**

```
SELECT
    Ingresso.id_ingresso,
    Ingresso.num_assento,
    Ingresso.preco,
    Ingresso.setor,
    Evento.local AS local_evento,
    Evento.data AS data_evento,
    Torcedor.nome AS nome_torcedor,
    Torcedor.pais AS pais_torcedor
FROM
    Ingresso
JOIN
    Evento ON Ingresso.id_evento = Evento.id_evento
JOIN
    Torcedor ON Ingresso.num_passaporte_torcedor = Torcedor.num_passaporte_torcedor
WHERE
    Ingresso.preco > 100.00
ORDER BY
    Ingresso.preco DESC;
```

**c) Seleção de Atletas com Detalhes de Comitê Olímpico Nacional e Internacional, Filtrados por Nome com LIKE**

**Junção de 3 ou mais tabelas, usando os operadores LIKE e BETWEEN:**

```
SELECT
    Atleta.nome AS nome_atleta,
    Atleta.nacionalidade,
    Atleta.altura,
    Atleta.peso,
    ComitêOlimpicoNacional.nome AS nome_con,
    ComitêOlimpicoInternacional.id_coi
FROM
    Atleta
JOIN
    ComitêOlimpicoNacional ON Atleta.id_con = ComitêOlimpicoNacional.id_con
JOIN
    ComitêOlimpicoInternacional ON ComitêOlimpicoNacional.id_coi =
    ComitêOlimpicoInternacional.id_coi
WHERE
    Atleta.nome LIKE 'Maria%'
ORDER BY
    Atleta.nome;
```

**d) Seleção de Eventos com Detalhes de Estádio e Modalidades, Filtrados por Localidade e Validade do Estádio**

**Junção de 3 ou mais tabelas, usando os operadores IN e IS NULL/IS NOT NULL:**

```
SELECT
    Evento.id_evento,
    Evento.local,
    Evento.data,
    Estádio.nome AS nome_estadio,
    Estádio.local AS local_estadio,
    Modalidades.nome AS nome_modalidade,
    Modalidades.tipo AS tipo_modalidade
FROM
    Evento
LEFT JOIN
    Estádio ON Evento.id_estadio = Estádio.id_estadio
JOIN
    Modalidades ON Evento.id_modalidade = Modalidades.id_modalidade
WHERE
    Evento.local IN ('Rio de Janeiro', 'Brasília')
    AND Estádio.id_estadio IS NOT NULL
ORDER BY
    Evento.data;
```

**e) Seleção da Altura Média dos Atletas por Nacionalidade, Ordenados por Altura Média**

**Junção de 2 ou mais tabelas com GROUP BY, sem HAVING, usando uma função (AVG). Use ORDER BY:**

```
SELECT
    Atleta.nacionalidade,
    AVG(Atleta.altura) AS altura_media
FROM
    Atleta
JOIN
    ComitOlimpicoNacional ON Atleta.id_con = ComitOlimpicoNacional.id_con
GROUP BY
    Atleta.nacionalidade
ORDER BY
    altura_media DESC;
```

**f) Seleção da Altura Média dos Atletas por Modalidade, Filtrados por Altura Média e Ordenados por Altura Média**

**Junção de 2 ou mais tabelas com GROUP BY e HAVING, usando uma função agregada (AVG):**

```
SELECT
    Modalidades.nome AS nome_modalidade,
    AVG(Atleta.altura) AS altura_media
FROM
    Atleta
JOIN
    Competicao ON Atleta.num_passaporte_atleta = Competicao.num_passaporte_atleta
JOIN
    Modalidades ON Competicao.id_modalidade = Modalidades.id_modalidade
GROUP BY
    Modalidades.nome
HAVING
    AVG(Atleta.altura) < 1.8
ORDER BY
    altura_media DESC;
```

**g) Seleção de Estádios com Capacidade Acima da Média**

**Subselect sem correlação:**

```
SELECT
    nome,
    capacidade
FROM
    Estadio
WHERE
    capacidade > (SELECT AVG(capacidade) FROM Estadio);
```

**h) Seleção de Atletas com Peso Acima da Média por Comitê Olímpico Nacional**

**Subselect com correlação:**

```
SELECT
    a.nome,
    a.peso,
    a.nacionalidade
FROM
    Atleta a
WHERE
    a.peso > (
        SELECT AVG(a2.peso)
        FROM Atleta a2
        WHERE a2.id_con = a.id_con
    );
```

**i) Seleção de Comitês Olímpicos Nacionais com Atletas Associados**

**Subselect com EXISTS:**

```
SELECT
    cn.id_con,
    cn.nome
FROM
    ComitêOlimpicoNacional cn
WHERE
    EXISTS (
        SELECT 1
        FROM Atleta a
        WHERE a.id_con = cn.id_con
    );
```

**j) Contagem de Atletas por Comitê Olímpico Nacional, com Mais de um Atleta Associado**

**Junção de 2 ou mais tabelas com GROUP BY e HAVING, usando COUNT:**

```
SELECT
    ComitêOlimpicoNacional.nome AS nome_con,
    COUNT(Atleta.num_passaporte_atleta) AS num_atletas
FROM
    Atleta
JOIN
    ComitêOlimpicoNacional ON Atleta.id_con = ComitêOlimpicoNacional.id_con
GROUP BY
    ComitêOlimpicoNacional.nome
HAVING
    COUNT(Atleta.num_passaporte_atleta) > 1
ORDER BY
    num_atletas DESC;
```

**Otimização de Consultas de Ingressos por Setor e Preço com Criação de Índice:**

```
CREATE INDEX IND_INGRESSOS  
ON ingresso(setor, preco)
```

**Otimização de Consultas de Esportes por Nome com Criação de Índice:**

```
CREATE INDEX IND_ESPORTE  
ON esportes(nome)
```

**Visualização dos Locais de Eventos com o Número Total de Ingressos Vendidos:**

```
CREATE VIEW INGRESSOS_MAIIS_VENDIDOS AS  
SELECT local, COUNT(id_ingresso) AS tot_ingressos  
FROM evento AS A JOIN ingresso AS B ON A.id_evento = B.id_evento  
GROUP BY LOCAL  
ORDER BY tot_ingressos DESC
```

**Visualização do Número de Atletas por Nacionalidade:**

```
CREATE VIEW ATLETAS_POR_NACIONALIDADE AS  
SELECT COUNT(num_passaporte_atleta), NACIONALIDADE FROM atleta  
GROUP BY NACIONALIDADE
```