MIDTERM CONTROL

Domain: Part - Supplier

PROGRAM TEXT

```
from dataclasses import dataclass


@dataclass
class Supplier:
    id: int
    name: str


@dataclass
class Part:
    id: int
    name: str
    cost: float
    supplier_id: int


@dataclass
class SupplierPart:
    supplier_id: int
    part_id: int


suppliers = [
    Supplier(1, "AutoParts LLC"),
    Supplier(2, "Ivanov IE - metals department"),
    Supplier(3, "Mechanics CJSC"),
    Supplier(4, "Industrial LLC"),
    Supplier(5, "Trade Company electronics department")
]

parts = [
    Part(1, "Crankshaft pulley", 1250.50, 1),
    Part(2, "Piston", 890.00, 1),
    Part(3, "Brake pad", 340.75, 1),
    Part(4, "Suspension lever", 1560.00, 2),
    Part(5, "Body panel", 2100.25, 2),
    Part(6, "Shock absorber", 980.00, 3),
    Part(7, "Nut M12", 15.50, 3),
    Part(8, "Oil filter", 250.00, 3),
    Part(9, "Spark plug", 120.00, 5),
    Part(10, "Oxygen sensor", 1850.00, 5)
]

supplier_parts = [
    SupplierPart(1, 1), SupplierPart(1, 2), SupplierPart(1, 3),
    SupplierPart(2, 4), SupplierPart(2, 5), SupplierPart(3, 6),
    SupplierPart(3, 7), SupplierPart(3, 8), SupplierPart(5, 9),
    SupplierPart(5, 10), SupplierPart(2, 6), SupplierPart(3, 1),
    SupplierPart(4, 2)
]


def query1():
    print("\n" + "=" * 80)
    print("QUERY 1: List of all related parts and suppliers (one-to-many)")
    print("=" * 80)
    result = sorted([(d, p) for p in suppliers for d in parts
                     if d.supplier_id == p.id],
                    key=lambda x: (x[1].id, x[0].name))
    for d, p in result:
        print(f"Supplier: {p.name:<35} | Part: {d.name:<25} | Cost: {d.cost:>10.2f}")
    return result


def query2():
    print("\n" + "=" * 80)
    print("QUERY 2: List of suppliers with total cost of parts")
    print("=" * 80)
```

```
    total = {d.supplier_id: sum(d2.cost for d2 in parts if d2.supplier_id == d.supplier_id)
             for d in parts}
    result = sorted([(p, total[p.id]) for p in suppliers
                     if p.id in total], key=lambda x: x[1])
    for p, s in result:
        print(f"Supplier: {p.name:<40} | Total cost: {s:>12.2f} rub.")
    return result


def query3():
    print("\n" + "=" * 80)
    print("QUERY 3: Suppliers with 'department' in name and their parts (many-to-many)")
    print("=" * 80)
    suppliers_with_dept = [p for p in suppliers if "department" in p.name.lower()]
    result = []
    for p in suppliers_with_dept:
        parts_p = [d for d in parts
                   if d.id in [sp.part_id for sp in supplier_parts
                               if sp.supplier_id == p.id]]
        if parts_p:
            print(f"\nSupplier: {p.name}")
            for d in parts_p:
                print(f"  ■■ Part: {d.name:<30} | Cost: {d.cost:>10.2f}")
            result.append((p, parts_p))
    return result


def create_pdf(output_buffer, filename="rk1pykyap_en.py"):
    try:
        from reportlab.lib.pagesizes import A4
        from reportlab.lib.units import mm
        from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer, Preformatted
        from reportlab.lib.styles import getSampleStyleSheet, ParagraphStyle
        from reportlab.lib.enums import TA_CENTER
        from reportlab.pdfbase import pdfmetrics
        from reportlab.pdfbase.ttfonts import TTFont
        import sys
        import os

        with open(filename, "r", encoding="utf-8") as f:
            code = f.read()

        doc = SimpleDocTemplate("rubegnyi_kontrol_en.pdf", pagesize=A4,
                                rightMargin=20 * mm, leftMargin=20 * mm,
                                topMargin=20 * mm, bottomMargin=20 * mm)

        styles = getSampleStyleSheet()

        code_style = ParagraphStyle('Code', parent=styles['Code'], fontName='Courier')
        heading_style = ParagraphStyle('Heading', parent=styles['Heading2'], fontName='Helvetica-Bold')
        title_style = ParagraphStyle('Title', parent=styles['Heading1'], fontSize=16,
                                     textColor='#0066CC', spaceAfter=10, alignment=TA_CENTER,
                                     fontName='Helvetica-Bold')

        story = []
        story.append(Paragraph("MIDTERM CONTROL", title_style))
        story.append(Paragraph("Domain: Part - Supplier", heading_style))
        story.append(Spacer(1, 10))

        story.append(Paragraph("PROGRAM TEXT", heading_style))
        story.append(Spacer(1, 5))
        story.append(Preformatted(code, code_style))

        story.append(Spacer(1, 10))
        story.append(Paragraph("EXECUTION RESULTS", heading_style))
        story.append(Spacer(1, 5))
        story.append(Preformatted(output_buffer, code_style))

        doc.build(story)
        print("\n✓ PDF created: rubegnyi_kontrol_en.pdf")
        return True
    except ImportError:
        print("\n■ Library reportlab not installed. PDF will not be created.")
        print("   Install: pip install reportlab")
        return False


import sys
from io import StringIO

if __name__ == "__main__":
```

```
    output_buffer = StringIO()
    old_stdout = sys.stdout
    sys.stdout = output_buffer

    print("\n" + "=" * 80)
    print("MIDTERM CONTROL")
    print("Domain: Part - Supplier")
    print("=" * 80)

    query1()
    query2()
    query3()

    print("\n" + "=" * 80)
    print("Program execution completed")
    print("=" * 80)

    output_text = output_buffer.getvalue()
    sys.stdout = old_stdout
    print(output_text)
    create_pdf(output_text, __file__)
```

EXECUTION RESULTS

```
================================================================================
MIDTERM CONTROL
Domain: Part - Supplier
================================================================================

================================================================================
QUERY 1: List of all related parts and suppliers (one-to-many)
================================================================================
Supplier: AutoParts LLC                       | Part: Brake pad                | Cost:     340.75
Supplier: AutoParts LLC                       | Part: Crankshaft pulley        | Cost:    1250.50
Supplier: AutoParts LLC                       | Part: Piston                   | Cost:     890.00
Supplier: Ivanov IE - metals department       | Part: Body panel               | Cost:    2100.25
Supplier: Ivanov IE - metals department       | Part: Suspension lever         | Cost:    1560.00
Supplier: Mechanics CJSC                      | Part: Nut M12                  | Cost:      15.50
Supplier: Mechanics CJSC                      | Part: Oil filter               | Cost:     250.00
Supplier: Mechanics CJSC                      | Part: Shock absorber           | Cost:     980.00
Supplier: Trade Company electronics department | Part: Oxygen sensor            | Cost:    1850.00
Supplier: Trade Company electronics department | Part: Spark plug               | Cost:     120.00

================================================================================
QUERY 2: List of suppliers with total cost of parts
================================================================================
Supplier: Mechanics CJSC                          | Total cost:      1245.50 rub.
Supplier: Trade Company electronics department    | Total cost:      1970.00 rub.
Supplier: AutoParts LLC                           | Total cost:      2481.25 rub.
Supplier: Ivanov IE - metals department           | Total cost:      3660.25 rub.

================================================================================
QUERY 3: Suppliers with 'department' in name and their parts (many-to-many)
================================================================================

Supplier: Ivanov IE - metals department
  ■■ Part: Suspension lever              | Cost:    1560.00
  ■■ Part: Body panel                    | Cost:    2100.25
  ■■ Part: Shock absorber                | Cost:     980.00

Supplier: Trade Company electronics department
  ■■ Part: Spark plug                    | Cost:     120.00
  ■■ Part: Oxygen sensor                 | Cost:    1850.00

================================================================================
Program execution completed
================================================================================
```