Parts table

Part (No, Description, Vendor, Address, UnitCost)

Functional dependencies
No ⇨ Description
Vendor ⇨ Address
No, Vendor ⇨ UnitCost

1NF
All values atomic

PartNo	Description	Vendor	Address	UnitCost
1234	Logic Chip	Fast Chips	Cupertino	10.00
1234	Logic Chip	Smart Chips	Phoenix	8.00
5678	Memory Chip	Fast Chips	Cupertino	3.00
5678	Memory Chip	Quality Chips	Austin	2.00
5678	Memory Chip	Smart Chips	Phoenix	5.00

Primary Key:
(No, Vendor)

2NF
Partial dependencies
PartNo ⇨ Description
Vendor ⇨ Address

Split table to become:
Part (PartNo, Description)
Vendor (Vendor, Address)
Cost (PartNo, Vendor, UnitCost)

3NF
Transitive dependencies
None

Final Tables:
Part (PartNo, Description)
Vendor (Vendor, Address)
Cost (PartNo, Vendor, UnitCost)

Part

PartNo	Description
1234	Logic Chip
5678	Memory Chip

Vendor

Vendor	Address
Fast Chips	Cupertino
Smart Chips	Phoenix
Quality Chips	Austin

PartNo	Vendor	UnitCost
1234	Fast Chips	10.00
1234	Smart Chips	8.00
5678	Fast Chips	3.00
5678	Quality Chips	2.00
5678	Smart Chips	5.00

Book Table: version 1

(BookTitle, Author, BookType, ListPrice, AuthorNationality, Publisher)

Functional dependencies
BookTitle ⇨ BookType, Publisher, ListPrice
Author ⇨ AuthorNationality
BookType ⇨ ListPrice
BookTitle, Author ⇨ BookType, Publisher, AuthorNationality, ListPrice

1NF
All values atomic

BookTitle	Author	BookType	ListPrice	Author Nationality	Publisher
ModernDB Management	Hoffer	Textbook	$100	American	Pearson
ModernDB Management	Prescott	Textbook	$100	Canadian	Pearson
ModernDB Management	McFadden	Textbook	$100	Canadian	Pearson
DB Systems	Elmasri	Textbook	$100	Canadian	Addison
DB Systems	Navathe	Textbook	$100	American	Addison
Harry Potter	JK Rowling	Fiction	$30	British	Bloomsbury

Primary Key:
(BookTitle, Author)

2NF
Partial dependancies
BookTitle ⇨ BookType, Publisher, ListPrice
Author ⇨ AuthorNationality

Split table to become:
Book: (BookTitle, BookType, Publisher, ListPrice)
Author (Author, AuthorNationality)
Written: (BookTitle, Author)

3NF
Transitive dependancies
BookType ⇨ ListPrice

Split table to become:
Book: (BookTitle, BookType, Publisher)
Price: (BookType, ListPrice)

Final Tables:
Book: (BookTitle, BookType, Publisher)
Author (Author, AuthorNationality)
Written: (BookTitle, Author)
Price: (BookType, ListPrice)

Book: (BookTitle, BookType, Publisher)
Author (Author, AuthorNationality)
Written: (BookTitle, Author)
Price: (BookType, ListPrice)

Book

BookTitle	BookType	Publisher
ModernDB Management	Textbook	Pearson
DB Systems	Textbook	Addison
Harry Potter	Fiction	Bloomsbury

Author

Author	Author Nationality
Hoffer	American
Prescott	Canadian
McFadden	Canadian
Elmasri	Canadian
Navathe	American
JK Rowling	British

Written

BookTitle	Author
ModernDB Management	Hoffer
ModernDB Management	Prescott
ModernDB Management	McFadden
DB Systems	Elmasri
DB Systems	Navathe
Harry Potter	JK Rowling

Price

BookType	ListPrice
Textbook	$100
Fiction	$30

Book Table: version 2

(BookTitle, Author, BookType, ListPrice, AuthorNationality, Publisher)

Functional dependencies
BookTitle ⇨ BookType, Publisher, ListPrice
Author ⇨ AuthorNationality
BookTitle, Author ⇨ BookType, Publisher, AuthorNationality, ListPrice

1NF
unchanged

2NF
unchanged

3NF
No transitive dependencies

Final Tables:
Book: (BookTitle, BookType, Publisher, ListPrice)
Author (Author, AuthorNationality)
Written: (BookTitle, Author)

Book

BookTitle	BookType	Publisher	List Price
ModernDB Management	Textbook	Pearson	$100
DB Systems	Textbook	Addison	$100
Harry Potter	Fiction	Bloomsbury	$30

Author

Author	Author Nationality
Hoffer	American
Prescott	Canadian
McFadden	Canadian
Elmasri	Canadian
Navathe	American
JK Rowling	British

Written

BookTitle	Author
ModernDB Management	Hoffer
ModernDB Management	Prescott
ModernDB Management	McFadden
DB Systems	Elmasri
DB Systems	Navathe
Harry Potter	JK Rowling

Grade table

Grade (ID, Name, Address, Major, UnitCode, Unit, Lecturer, Office, Grade)

Functional dependencies
ID ⇨ Name, Address, Major
UnitCode ⇨ Unit, Lecturer, Office
Lecturer ⇨ Office
ID, UnitCode ⇨ Grade, Name, Address, Major, Unit, Lecturer, Office

1NF
All values atomic

Primary Key:
(ID, UnitCode)

2NF
Partial dependencies
ID ⇨ Name, Address, Major
UnitCode ⇨ Unit, Lecturer, Office

Split table to become:
Student (ID, Name, Address, Major)
Unit (Unit, Lecturer, Office)
Grade (ID, Unit, Grade)

3NF
Transitive dependencies
Lecturer ⇨ Office

Split table to become:
Unit (Unit, Lecturer)
Lecturer (Lecturer, Office)

Final Tables:
Student (ID, Name, Address, Major)
Unit (Unit, Lecturer)
Lecturer (Lecturer, Office)
Grade (ID, Unit, Grade)

Invoice table not structured

CustABN	CustName	CustAddress	CustSuburb	InvoiceNo	InvDate	ItemNo	ItemDesc	ItemPrice	Quantity

Functional dependencies

CustABN ⇨ CustName, CustAddress, CustSuburb

InvoiceNo ⇨ InvDate, CustABN, CustName, CustAddress, CustSuburb

ItemNo ⇨ ItemDesc, ItemPrice

InvoiceNo, ItemNo ⇨ Quantity, InvDate, CustABN, CustName, CustAddress, CustSuburb, ItemDesc, ItemPrice

1NF

Table is in 1NF (all *atomic values*)

Primary key: InvoiceNo, ItemNo

2NF

Partial dependencies:

InvoiceNo ⇨ InvDate, CustABN, CustName, CustAddress, CustSuburb

ItemNo ⇨ ItemDesc, ItemPrice

Split table to become:

Invoice

CustABN	CustName	CustAddress	CustSuburb	InvoiceNo	InvDate

Item

ItemNo	ItemDesc	ItemPrice

InvoiceItem

InvoiceNo	ItemNo	Quantity

3NF

Transitive dependencies

CustABN ⇨ CustName, CustAddress, CustSuburb

Split table to become:

Customer

CustABN	CustName	CustAddress	CustSuburb

Invoice

InvoiceNo	InvDate	CustABN

Final Tables:

Customer

CustABN	CustName	CustAddress	CustSuburb

Invoice

InvoiceNo	InvDate	CustABN

Item

ItemNo	ItemDesc	ItemPrice

InvoiceItem

InvoiceNo	ItemNo	Quantity

Student Tables

Student (No, Name, Course, CourseName, CoordId, CoordName)

Functional dependencies
No ⇨ Name
Course ⇨ CouseName
CoordId ⇨ CoordName
No, Course, CoordId ⇨ Name, CourseName, CoordName

1NF
All values atomic

No	Name	Course	CourseName	CoordId	CoordName
783917	Charlie Chaplin	BIT99	Bach of IT	23	Fred Flintstone
983403	Pluto	BIT99	Bach of IT	23	Fred Flintstone
983403	Pluto	Cert IV	Cert 4 of IT	48	Donald Duck
983403	Pluto	Cert IV	Cert 4 of IT	31	Minnie Mouse
100234	Huey	BIT3	Bach of IT	45	Mary Poppins
100234	Huey	BIT3	Bach of IT	48	Donald Duck
100234	Huey	Cert IV	Cert 4 of IT	48	Donald Duck
100234	Huey	Cert IV	Cert 4 of IT	31	Minnie Mouse

Primary Key:
(No, Course, CoordId)

2NF
Partial dependencies
No ⇨ Name
Course ⇨ CouseName
CoordId ⇨ CoordName

Split table to become:
Student (No, Name)
Course (Course, CouseName)
Coordinator(CoordId, CoordName)
Studying (No, Course, CoordId)

3NF
Transitive dependencies
None

Final Tables:
Student (No, Name)
Course (Course, CouseName)
Coordinator(CoordId, CoordName)
Studying (No, Course, CoordId)