1. Simplify each sentence and use truth table to validate.
(a) Smoke ⇒ Smoke = ¬Smoke ∨ Smoke

Smoke	**KB**	**Model of KB True?**
True	False ∨ True	Yes
False	True ∨ False	Yes

Therefore, this sentence is always **valid**.

(b) Smoke ⇒ Fire = ¬Smoke ∨ Fire

Smoke	**Fire**	**KB**	**Model of KB True?**
True	True	False ∨ True	Yes
False	True	True ∨ True	Yes
True	False	False ∨ False	No
False	False	True ∨ False	Yes

KB is not true in the third model, so this sentence is satisfiable but not valid. The answer is **neither**.

(c) Smoke ∨ Fire ∨ ¬Fire

Smoke	**Fire**	**KB**	**Model of KB True?**
True	True	False ∨ True ∨ False	Yes
False	True	True ∨ True ∨ False	Yes
True	False	False ∨ False ∨ True	Yes
False	False	True ∨ False ∨ True	Yes

Therefore, this sentence is always **valid**.

(d) (Smoke ⇒ Fire) ⇒ (¬Smoke ⇒ ¬Fire) = (¬Smoke ∨ Fire) ⇒ (¬¬Smoke ∨ ¬Fire)
= (¬ (¬Smoke ∨ Fire) ∨ (Smoke ∨ ¬Fire)) = (Smoke ∧ ¬ Fire) ∨ Smoke ∨ ¬Fire

Smoke	**Fire**	**KB**
True	True	True
False	True	False
True	False	True
False	False	True

KB is not true in the second model, so this sentence is satisfiable but not valid. The answer is **neither**.

(e) (Smoke ⇒ Fire) ⇒ ((Smoke ∨ Heat) ⇒ Fire) = (¬Smoke ∨ Fire) ⇒ (¬(Smoke ∨ Heat) ∨ Fire)
= (¬ (¬Smoke ∨ Fire) ∨ ((¬Smoke ∧ ¬Heat) ∨ Fire))

= (Smoke ∧ ¬Fire) ∨ (¬Smoke ∧ ¬Heat) ∨ Fire

Smoke	**Fire**	**Heat**	**KB**
True	True	True	True
False	True	True	True
True	False	True	True
False	False	True	False
True	True	False	True
False	True	False	True
True	False	False	True
False	False	False	True

KB is not true in the third model, so this sentence is satisfiable but not valid. The answer is **neither**.

(f) ((Smoke ∧ Heat) ⇒ Fire) ⇔ ((Smoke ⇒ Fire) ∨ (Heat ⇒ Fire))
= (¬(Smoke ∧ Heat) ∨ Fire) ⇔ ((¬Smoke ∨ Fire) ∨ (¬ Heat ∨ Fire))
= (¬Smoke ∨ ¬Heat ∨ Fire) ⇔ (¬Smoke ∨ Fire ∨ ¬ Heat)
If we let P = (¬Smoke ∨ ¬Heat ∨ Fire), this sentence becomes P⇔P = (P ⇒ P) ∧ (P ⇒P) = (P ⇒ P) = (¬P∨P), which we proved in part (a) that it is valid. The truth table is as follows:

Smoke	**Fire**	**Heat**	**KB**
True	True	True	True
False	True	True	True
True	False	True	True
False	False	True	True
True	True	False	True
False	True	False	True
True	False	False	True
False	False	False	True

Therefore, this sentence is always **valid**.

2.

(a) P(A, B, B), P(x, y, z)

Θ= {x/A, y/B, z/B}

(b) Q(y, G(A, B)), Q(G(x, x), y).

There is no such unifier. If we have to assign y to both G(x, x) and G(A, B), but there is no way to unify G(x, x) and G(A, B), since x cannot both represent A and B.

(c) Older(Father(y), y), Older(Father(x), John).

Θ= {x/John, y/John}

(d) Knows(Father(y), y), Knows(x, x).

This is no such unifier, since we cannot unify Father(y) and y.

3.

(a)

A x, food(x) => likes (John, x)

food(Apples)

food(Chicken)

A x, y, eats(x,y) & -killed(y, x) => food(y)

A x (E y, killed(x, y)) => -alive(x)

eats(Bill, peanuts) & alive(Bill)

A x, eats(Bill, x) => eats(Sue, x)

(b)

1. A x, food(x) => likes (John, x) = **(-food(x) | likes(John, x))**

2. food(Apples)

3. food(Chicken)

4. A x, y, eats(x,y) & -killed(x, y) => food(y)

= (- (eats(x,y) & -killed(x, y)) | food(y)

= **-eats(x,y) | killed(x, y) | food(y)**

5. A x (E y, killed(x, y)) => -alive(x) = **-killed(x,y) | -alive(x)**

6. eats(Bill, peanuts)

7. alive(Bill)

8. A x, eats(Bill, x) => eats(Sue, x) = **-eats(Bill, x) | eats(Sue, x)**

(c) Using resolution and contradiction:

Proposition: -**likes(John, peanuts)**

Perform resolution on sentence 4 and 5, we have:

9. -eats(x,y) | -alive(x) | food(y)

Perform resolution on sentence 7,9 with the unification {x/Bill}, we have:

10. -eats(Bill, y) | food(y)

Perform resolution on sentence 6,10 with the unification {y/peanuts}, we have:

11. food(peanuts)

Perform resolution on sentence 1,11 with the unification {x/peanuts}, it follows that:

12. likes(John, peanuts)

Unifying 12 with the proposition, we have the empty clause. So we reach the conclusion that John likes peanuts.

(d) Using resolution and contradiction:

The problem is E y, food(y) & eats(Sue, y)

Proposition: **-food(y) | -eats(Sue, y)**.

Resolve proposition and sentence 8 with {x/y}, we have:

13. -eats(Bill, x) | -food(x)

Resolve sentence 6,13 with {x/peanuts}, we have:

14. -food(peanuts)
Resolve sentence 11, 14 and we reach the empty clause.
So with the result from unification, y is peanuts, and we reach the conclusion that Sue eats peanuts.

(e) We replace 6 with the new axioms:
15. A x (E y, -eats(x,y)) => die(x) = **eats(x,y) | die(x)**
16. A x, die(x) => -alive(x) = **-die(x) | -alive(x)**
And 7 remains unchanged.
The problem is E y, food(y) & eats(Sue, y)
Proposition: **-food(y) | -eats(Sue, y)**.
Perform resolution on proposition,8, we have:
17. **-eats(Bill, y) | -food(y)**
Perform resolution on 15,17 with the unification {x/Bill}, we have:
18. die(Bill) | -food(y)
Perform resolution on 16,18 with the unification {x/Bill}, we have:
19. -alive(Bill) | -food(y)
Perform resolution on 7,19, we have:
20. -food(y)
Perform resolution on 10, 20, we have:
21. -eats(Bill, y)
Perform resolution on 15,21 with {x/Bill}, we have
22. die(Bill)
Resolve 16,22 with {x/Bill}, we have:
23. -alive(Bill)
Resolve 7,23 and we get the empty clause.
So from the substitution of y, we can conclude that Sue eats everything that Bill eats.

4.
Symbols:
Mythical: The unicorn is mythical.
Immortal: The unicorn is immortal.
Mammal: The unicorn is mammal.
Horned: The unicorn is horned.

(a)
mythical $\Rightarrow$ immortal
$\neg$mythical $\Rightarrow \neg$ immortal $\wedge$ mammal
immortal $\vee$ mammal $\Rightarrow$ horned
horned $\Rightarrow$ magical

(b)
1. $\neg$mythical $\vee$ immortal
2. $\neg$mythical $\Rightarrow \neg$ immortal $\wedge$ mammal = mythical $\vee$ ($\neg$ immortal $\wedge$ mammal)
= (mythical $\vee$ $\neg$ immortal) $\wedge$ (mythical $\vee$ mammal)
3. immortal $\vee$ mammal $\Rightarrow$ horned = ($\neg$ immortal $\wedge$ $\neg$ mammal) $\vee$ horned
= ($\neg$ immortal $\vee$ horned) $\wedge$ ($\neg$ mammal $\vee$ horned)
4. $\neg$horned $\vee$ magical

(c)
KB:
1. $\neg$mythical $\vee$ immortal
2. mythical $\vee$ $\neg$ immortal
3. mythical $\vee$ mammal
4. $\neg$ immortal $\vee$ horned
5. $\neg$ mammal $\vee$ horned
6. $\neg$horned $\vee$ magical

Mythical:
Add $\neg$mythical to the knowledge base.
Resolve $\neg$mythical with 3, then we get $\neg$ mammal, and there are no more steps to perform. Therefore, we cannot prove that the unicorn is mythical.

Magical & horned:
Resolve 1, 3, we have:
7. immortal $\vee$ mammal
Resolve 4,7, we have:
8. horned $\vee$ mammal
Resolve 5,8, we have:

9. horned
Resolve 6,9, we have:
10. magical
So this KB entails that the unicorn is horned and magical.

5.

(a)

First, we want to show that α is valid if True$|=\alpha$.

We know that a sentence is valid if it is true in all models.

By the definition of entailment, True $|= \alpha$ if and only if, in every model in which True is true, α is also true. Since, True is true in every model, α is true in every model. Then by the definition of validity, α is valid.

Then, we show that True$|=\alpha$ if α is valid.

α being valid means that α is true in every model.

True $|= \alpha$ if and only if, in every model in which True is true, α is also true. Since True is true in every model and α is true in every model, this statement is proven to be true.

(b)

By the definition of entailment, False$|= \alpha$ if and only if, in every model in which False is true, α is also true; moreover, $M(\text{False}) \subseteq M(\alpha)$. Since no model makes False to be true, $M(\text{False}) = \emptyset$. We know that for any α and $M(\alpha)$, $\emptyset \subseteq M(\alpha)$, so $M(\text{False}) \subseteq M(\alpha)$ and False$|= \alpha$ is true.

(c)

$\alpha |= \beta$ if and only if, in every model in which α is true, β is also true.

First we prove the statement in forward direction.

If $\alpha |= \beta$, then $\alpha \Rightarrow \beta$ is true for all models where α is true, since in this case β must be also true. And $\alpha \Rightarrow \beta$ is also true for all models where α is false by definition of implication. So $\alpha \Rightarrow \beta$ is valid in such scenario.

Then we prove the backward direction.

If $\alpha \Rightarrow \beta$ is true, then if α is true, β is true, and it follows that $\alpha |= \beta$. If α is false, by part(b), we can also conclude that $\alpha |= \beta$.

(d)

By the implication elimination law, we know that $(\alpha \Rightarrow \beta) \equiv (\neg\alpha \vee \beta)$. And we also know that α is valid iff $\neg\alpha$ is unsatisfiable, $(\alpha \Rightarrow \beta)$ is valid if and only if $\neg(\neg\alpha \vee \beta)$ is unsatisfiable. Then, by the De Morgan Law, we know $\neg(\neg\alpha \vee \beta) \equiv (\alpha \wedge \neg\beta)$. Thus we have $\alpha |= \beta$ if and only if the sentence $\alpha \wedge \neg\beta$ is unsatisfiable.

Another way of proving this:

First we prove the forward direction by contradiction.

Suppose $\alpha |= \beta$. Assume that $\alpha \wedge \neg\beta$ is satisfiable, then in this case, α must be true and β must be false. However, we know that $\alpha |= \beta$, which means if α is true, β has to be true. A contradiction. Therefore, $\alpha |= \beta$ means $\alpha \wedge \neg\beta$ is unsatisfiable

Then we prove the backward direction.

Given $\alpha \land \neg\beta$ is unsatisfiable. Then, if α is true, $\neg\beta$ must be false and β must be true. The definition of entailment says in all worlds where α is true, β is true. And we already proved that if α is true, then β must be true.