

1.

分析:

将前提假设 $P \vee Q$ 拆分成两个假设 HP 和 HQ 。后面的逻辑析取，也可以通过 `left` 和 `right` 的 tactic，分别用两个假设证明对应的部分即可。

代码:

```
Theorem or_commut : forall P Q : Prop,  
  P \vee Q -> Q \vee P.  
Proof.  
  intros P Q [HP | HQ].  
  - right. apply HP.  
  - left. apply HQ.  
Qed.
```

运行结果:

```
(** **** Exercise: 1 star, standard (or_commut) *)  
Theorem or_commut : forall P Q : Prop,  
  P \vee Q -> Q \vee P.  
Proof.  
  intros P Q [HP | HQ].  
  - right. apply HP.  
  - left. apply HQ.  
Qed.  
(** [] *)
```

2.

分析:

先将 \sim 用 `unfold` 展开， $\sim P$ 就变成了 $P \rightarrow \text{False}$ ，整个命题变成了 $P \wedge (P \rightarrow \text{False}) \rightarrow \text{False}$ 。分别用 $P \rightarrow \text{False}$ 和 P 去 `apply` 即可。

代码:

```
Theorem not_both_true_and_false : forall P : Prop,  
  ~ (P /\ ~P).  
Proof.  
  unfold not.  
  intros P [HP NP].  
  apply NP. apply HP.  
Qed.
```

运行结果：

```
(** **** Exercise: 1 star, standard (not_both_true_and_false) *)
Theorem not_both_true_and_false : forall P : Prop,
  ~ (P /\ ~P).
Proof.
  unfold not.
  intros P [HP NP].
  apply NP. apply HP.
Qed.
```

3&4.

`Logic.v` 中的 `Theorem or_distributes_over_and` 是3和4的综合，这里就按照文件里，一起证明了。

分析：

`intros` 之后，先用 `split` 把"iff"问题转化为两个子问题。

先证明3（从左到右）。`intros [HP | [HQ HR]]` 将 `P Q R` 条件分别命名。又分为两个子问题，第一个是 `HP` 证明右边，第二个是 `HQ HR` 证明右边。第一个情况，又用 `split` 将要证明的结论分为两个部分。注意到两个部分都是析取且都有 `P`，直接对 `left` `apply HP` 即可；第二个情况，与前面相反。对 `right` 分别 `apply HQ` 和 `apply HR`。

再证明4（从右到左）。`intros` 四个假设，注意是析取还是合取。分四种情况，证明的手段跟前面相似。

代码：

```
Theorem or_distributes_over_and : forall P Q R : Prop,
  P \/ (Q /\ R) <=> (P \/ Q) /\ (P \/ R).
Proof.
  intros. split.
  - intros [HP | [HQ HR]].
    * split.
      + left. apply HP.
      + left. apply HP.
    * split.
      + right. apply HQ.
      + right. apply HR.
  - intros [[HP | HQ] [HP' | HR]].
    * left. apply HP.
    * left. apply HP.
    * left. apply HP'.
    * right. split. apply HQ. apply HR.
Qed.
```

运行结果：

```
Theorem or_distributes_over_and : forall P Q R : Prop,  
  P \/\ (Q /\ R) <=> (P \/\ Q) /\ (P \/\ R).
```

Proof.

```
  intros. split.  
- intros [HP | [HQ HR]].  
  * split.  
    + left. apply HP.  
    + left. apply HP.  
  * split.  
    + right. apply HQ.  
    + right. apply HR.  
- intros [[HP | HQ] [HP' | HR]].  
  * left. apply HP.  
  * left. apply HP.  
  * left. apply HP'.  
  * right. split. apply HQ. apply HR.
```

Qed.