Adding and removing character devices in a Linux system typically involves managing the associated kernel module and creating/removing the device node in the `/dev` directory.

Adding a Character Device:

- **Build the Driver:** Compile the character device driver source code (e.g., `my_char_driver.c`) to generate the kernel module (`.ko` file). This is usually done using a `Makefile`.
  Code
  ```
  make
  ```
- **Insert the Module:** Load the compiled kernel module into the kernel using `insmod`.
  Code
  ```
  sudo insmod my_char_driver.ko
  ```
- **Create the Device Node:** Create the character device node in `/dev` using `mknod`. You will need the major and minor numbers assigned to your device by the kernel (which can be found in `/proc/devices` after module insertion).
  Code
  ```
  sudo mknod /dev/my_char_device c <major_number> <minor_number>
  ```

  c indicates a character device and <major_number> and <minor_number> are the device's identifiers.
- **Set Permissions (Optional):** Adjust permissions on the device node as needed using `chmod`.
  Code
  ```
  sudo chmod 666 /dev/my_char_device
  ```

Removing a Character Device:

- **Remove the Device Node:** Delete the device node from `/dev` using `rm`.
  Code
  ```
  sudo rm /dev/my_char_device
  ```
- **Remove the Module:** Unload the kernel module from the kernel using `rmmod`.
  Code
  ```
  sudo rmmod my_char_driver
  ```

Important Notes:

- **Major/Minor Numbers:**

  Character devices are identified by a major number (identifying the driver) and a minor number (identifying a specific device instance handled by that driver). These are typically assigned dynamically by the kernel during driver registration.
- **Error Handling:**

  In a real-world scenario, robust error handling should be implemented in the driver code to manage device registration and unregistration.

- **udev:**
  For automatic device node creation and permission management, `udev` can be configured to handle character devices upon module insertion. This is often preferred in production environments.