

```
# 1.`send_message`: Отправляет текстовое сообщение пользователю.
# bot.send_message(chat_id, text)

# 2.`reply_to`: Является способом отправки ответа на сообщение
пользователя.
# bot.reply_to(message, text)

# 3.`forward_message`: Пересылает сообщение от одного пользователя
другому.
# bot.forward_message(to_chat_id, from_chat_id, message_id)

# 4.`send_photo`: Отправляет фотографию.
# bot.send_photo(chat_id, photo, caption=None)

# 5.`send_audio`: Отправляет аудиофайл.
# bot.send_audio(chat_id, audio)

# 6.`send_document`: Отправляет документ или любой другой тип
файла.
# bot.send_document(chat_id, data)

# 7.`send_video`: Отправляет видеофайл.
# bot.send_video(chat_id, data)

# 8.`send_sticker`: Отправляет стикер.
# bot.send_sticker(chat_id, sticker)

# 9.`send_location`: Отправляет геолокацию.
# bot.send_location(chat_id, latitude, longitude)

# 10.`message_handler`: Декоратор для обработки входящих сообщений
с различными фильтрами, включая команды, регулярные выражения и
так далее.
# @bot.message_handler(commands=['start', 'help'])

# 11.`inline_handler`: Декоратор для обработки входящих inline
query.
# @bot.inline_handler(lambda query: query.query == 'text')

# 12.`callback_query_handler`: Декоратор для обработки callback
запросов от inline кнопок.
# @bot.callback_query_handler(func=lambda call: True)

# 13.`polling`: Запускает бота, постоянно опрашивая серверы
Telegram на предмет новых сообщений.
# bot.polling(none_stop=True)
```

```
# 14. `register_next_step_handler`: Регистрирует функцию, которая  
функцию, которая будет вызвана сразу после получения следующего  
сообщения от пользователя.  
# bot.register_next_step_handler(message, function)
```