

DaisyThread

- *run* (A) vykonává kód vlákna
- *onExit* (A) podtřídy mohou provést „úklid“ prostředků, voláno při ukončení činnosti vlákna
- *nice* touto metodou se vlákno „vzdá“ procesoru, protože čeká na událost
- *getReturnCode* 0-vlákno ukončeno korektně, -1 běží, >0 chybový kód po skončení vlákna

DaisyTCPListener

- *konstruktor* (obdrží otevřený socket)
- *listen* (A) provádí zpracování přijímaných dat
- *onExit* zavře socket
- *receiveData* přečte ze socketu data, která přicházejí od klienta
- *sendData* odešle data klientovi
- *run* přinejmenším zavolá *listen*

DaisyTCPListenerFactory

- *konstruktor* (obdrží číslo TCP portu)
- *create* (A) (obdrží otevřený socket) vytvoří instanci podtřídy *DaisyTCPListener*
- *onExit* A
- *run* čeká na příchozí spojení a poté zavolá *create*, pak čeká na další příchozí spojení

DaisyHTTPServer

- *konstruktor* (obdrží otevřený socket)
- *listen* načte od klienta http příkaz (GET, POST), hlavičky, případně tělo požadavku a předá metodě *get* (již rozparované na jednotlivé části)
- *get* A zpracuje jeden HTTP požadavek
- *answer* (obdrží kód odpovědi HTTP a tělo odpovědi) sestaví HTTP odpověď a pošle klientovi

DaisyHTML

tato třída obsahuje metody pro snažší vytváření HTML-kódu a odstíní programátora třídy *DaisyHTTPTestServer* od nutnosti znát podrobně HTML, navíc např. záměnou implementace může být místo HTML generováno WML, XHTML apod.

DaisyHTTPTestServer (např. **DaisyScitani**)

- *konstruktor* (obdrží otevřený socket)
- *get* rozpozná dle cesty v URL stránku (např. / a /plus) a vytvoří odpovídající HTML kód

DaisyHTTPTestServerFactory (např. **DaisyScitaniFactory**)

- *konstruktor* (obdrží číslo TCP portu)
- *create* (obdrží otevřený socket) založí instanci *DaisyHTTPTestServer* a zavolá *run*

Hlavní program založí instanci *DaisyHTTPTestServerFactory* a zavolá *run*.