# Application Insights Resources

**Get started** (~5 minutes to read):

<https://docs.microsoft.com/azure/application-insights/app-insights-overview>

**Instrument a running web app without code changes**

This page describes how to add instrumentation to a running web app on IIS (~7 minutes to read the whole page, but the IIS piece is a subset):

[https://docs.microsoft.com/azure/application-insights/app-insights-monitor-performance-live-website-now#monitor-a-live-iis-web-app](https://docs.microsoft.com/azure/application-insights/app-insights-monitor-performance-live-website-now" \l "monitor-a-live-iis-web-app)

Important, please note:

1. This requires restarting IIS (not the whole server).
2. The app should also be instrumented for App Insights; publishing updated versions of the app after “live” enablement can disable live-added instrumentation.

**Instrument a web app at development/compile time**

This page describes how to add Application Insights to an app at compile time (~6 minutes to read).

[https://docs.microsoft.com/azure/application-insights/app-insights-asp-net](https://docs.microsoft.com/azure/application-insights/app-insights-asp-net-trace-logs)

This does not modify the app’s code; it adds a config file and Nuget packages.

**Do more**

1. In addition to the Azure portal, its telemetry can be surfaced through [Operations Management Suite](https://www.microsoft.com/cloud-platform/operations-management-suite) alongside other operational telemetry and monitoring for a “single pane of glass”. See <https://blogs.technet.microsoft.com/msoms/2016/09/26/application-insights-connector-in-oms/>
2. Developers can add code to [send custom events](https://docs.microsoft.com/azure/application-insights/app-insights-api-custom-events-metrics), in addition to what Application Insights gets “out of box”/without code modification.
3. If developers are already using NLog, log4Net, or System.Diagnostics.Trace, those logs [can be sent to Application Insights](https://docs.microsoft.com/azure/application-insights/app-insights-asp-net-trace-logs) without code changes.
4. Telemetry (raw or query output) can be exported to [Power BI for dashboarding](https://docs.microsoft.com/azure/application-insights/app-insights-export-power-bi), or [to Azure storage in JSON format](https://docs.microsoft.com/azure/application-insights/app-insights-export-telemetry) for further analysis in other tools

**Advanced**

As an app scales and sends very large volumes of telemetry, developers may decide to implement custom logic for telemetry data reduction, such as sampling or aggregation.

See the MSDN article [DevOps – Optimize Telemetry with Application Insights](https://msdn.microsoft.com/magazine/mt808502) and the associated [sample app github repo](https://github.com/Azure-Samples/application-insights-dotnet-data-reduction).